

AD-A043 559

DAVID W TAYLOR NAVAL SHIP RESEARCH AND DEVELOPMENT CE--ETC F/G 9/2
COMPUTER CENTER REFERENCE MANUAL, REVISION.(U)

JUN 77 S E GOOD, D V SOMMER

UNCLASSIFIED

CMLD-77-11

NL

1 OF 3
AD
A043 559



IAU NC
DDC FILE COPY
COMPUTER CENTER REFERENCE MANUAL

CMLD-77-11

ADA 043559

DAVID W. TAYLOR NAVAL SHIP RESEARCH AND DEVELOPMENT CENTER

Bethesda, Md. 20084



COMPUTER CENTER REFERENCE MANUAL

by

Sharon E. Good

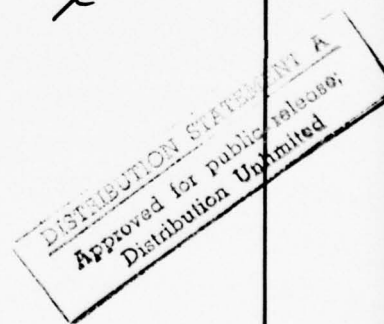
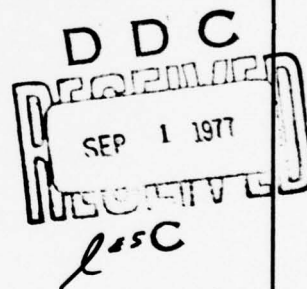
&

David V. Sommer

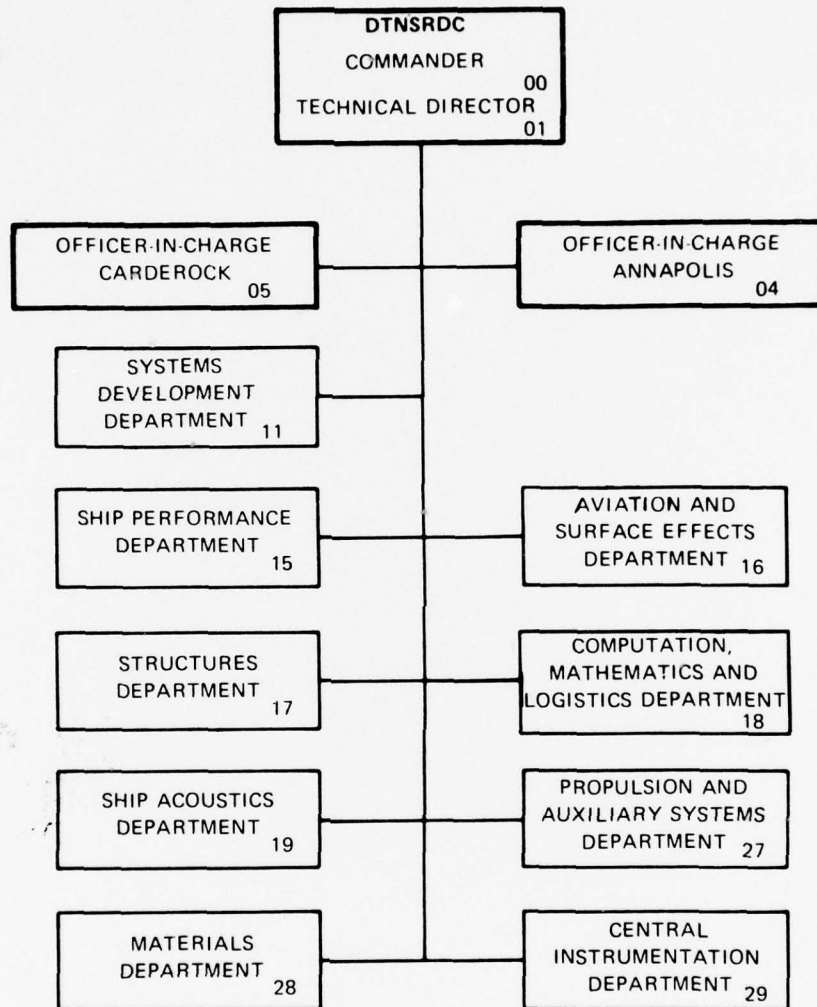
✓ Computation, Mathematics and
Logistics Department
(18)

June 1977

CMLD-77-11



MAJOR DTNSRDC ORGANIZATIONAL COMPONENTS



UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

| REPORT DOCUMENTATION PAGE | | READ INSTRUCTIONS BEFORE COMPLETING FORM |
|--|--|---|
| 1. REPORT NUMBER 14 CMLD-77-11 ✓ | 2. GOVT ACCESSION NO. | 3. RECIPIENT'S CATALOG NUMBER |
| 4. TITLE (and Subtitle) 6 Computer Center Reference Manual. Revision. | 5. TYPE OF REPORT & PERIOD COVERED 9 Final rept. | 6. PERFORMING ORG. REPORT NUMBER |
| 7. AUTHOR(s) 10 Sharon E./Good David V./Sommer | 8. CONTRACT OR GRANT NUMBER(s) | |
| 9. PERFORMING ORGANIZATION NAME AND ADDRESS DTNSRDC, User Services, Code 1892 Bethesda, Maryland 20084 | 10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS | |
| 11. CONTROLLING OFFICE NAME AND ADDRESS | 12. REPORT DATE 11 June 1977 | 13. NUMBER OF PAGES 273 |
| 14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) 12 276p. | 15. SECURITY CLASS. (of this report) UNCLASSIFIED | 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE |
| 16. DISTRIBUTION STATEMENT (of this Report) Approved For Public Release; DISTRIBUTION UNLIMITED | | |
| 17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report) | | |
| 18. SUPPLEMENTARY NOTES | | |
| 19. KEY WORDS (Continue on reverse side if necessary and identify by block number) CDC 6000 Control Card NOS/BE Operating System Software Documentation Interactive Processing Computer Remote Job Entry Graphics Programming Languages Hardware | | |
| 20. ABSTRACT (Continue on reverse side if necessary and identify by block number) The Computer Center Reference Manual provides an introduction to the CDC 6000 NOS/BE operating system for applications programmers. Some information has been distilled from many individual documents and augmented to reflect usage at DTNSRDC. Control card examples and descriptions of hardware and software are included. | | |

DD FORM 1 JAN 73 1473

EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

406 847

DAVID W. TAYLOR
NAVAL SHIP RESEARCH AND DEVELOPMENT CENTER
BETHESDA, MARYLAND 20084

*
*
* COMPUTER CENTER REFERENCE MANUAL *
*
*

BY
DAVID V SOMMER
SHARON E GOOD
USER SERVICES BRANCH

CODE 1892

PHONE (202) 227-1907 (301) 267-3343
AUTOVON 287-1907 281-3343
IDS 150-1907 1229-3343

FOR RECORDED MESSAGE ON COMPUTER STATUS (301) 229-6622

QUESTIONS AND REQUESTS FOR MORE DETAILED INFORMATION
SHOULD BE DIRECTED TO CODE 1892, BLDG. 17, RM. 100.

COMPUTATION, MATHEMATICS AND LOGISTICS DEPARTMENT
DEPARTMENTAL REPORT

JUNE 1977

CMLD-77-11

.. THROUGH REVISION 0 (JUNE 1977) ..

| | |
|---------------------------------|---|
| ACCESSION for | |
| NTIS | White Section <input checked="" type="checkbox"/> |
| IDS | Blue Section <input type="checkbox"/> |
| UNANNOUNCED | <input type="checkbox"/> |
| JUSTIFICATION | |
| BY | |
| DISTRIBUTION/AVAILABILITY CODES | |
| Dist | ALL <input type="checkbox"/> FOR SPECIAL <input type="checkbox"/> |
| A | |

JUNE 1977

REVISION RECORD

PAGE R-1

| REVISION | DESCRIPTION |
|------------|--------------------|
| 0 (JUN 77) | ORIGINAL PRINTING. |

JUNE 1977

PAGE R-2

*** LIST OF EFFECTIVE PAGES ***

| PAGE | REV | DATE | PAGE | REV | DATE |
|------|-----|-----------|--------|-----|-----------|
| R-1 | 0 | JUNE 1977 | 3-3 | 0 | JUNE 1977 |
| R-2 | 0 | JUNE 1977 | 3-4 | 0 | JUNE 1977 |
| R-3 | 0 | JUNE 1977 | 3-5 | 0 | JUNE 1977 |
| R-4 | 0 | JUNE 1977 | 3-6 | 0 | JUNE 1977 |
| I | 0 | JUNE 1977 | 3-7 | 0 | JUNE 1977 |
| II | 0 | JUNE 1977 | 3-8 | 0 | JUNE 1977 |
| III | 0 | JUNE 1977 | 3-9 | 0 | JUNE 1977 |
| IV | 0 | JUNE 1977 | 3-10 | 0 | JUNE 1977 |
| V | 0 | JUNE 1977 | 3-11 | 0 | JUNE 1977 |
| VI | 0 | JUNE 1977 | 3-12 | 0 | JUNE 1977 |
| VII | 0 | JUNE 1977 | 3-13 | 0 | JUNE 1977 |
| VIII | 0 | JUNE 1977 | 3-14 | 0 | JUNE 1977 |
| IX | 0 | JUNE 1977 | 3-15 | 0 | JUNE 1977 |
| 1-1 | 0 | JUNE 1977 | 3-16 | 0 | JUNE 1977 |
| 1-2 | 0 | JUNE 1977 | 3-17 | 0 | JUNE 1977 |
| 1-3 | 0 | JUNE 1977 | 3-18 | 0 | JUNE 1977 |
| 1-4 | 0 | JUNE 1977 | 3-19 | 0 | JUNE 1977 |
| 1-5 | 0 | JUNE 1977 | 3-20 | 0 | JUNE 1977 |
| 1-6 | 0 | JUNE 1977 | 4-1 | 0 | JUNE 1977 |
| 1-7 | 0 | JUNE 1977 | 4-2 | 0 | JUNE 1977 |
| 1-8 | 0 | JUNE 1977 | 4-3 | 0 | JUNE 1977 |
| 1-9 | 0 | JUNE 1971 | 4-4 | 0 | JUNE 1977 |
| 1-10 | 0 | JUNE 1977 | 4-5 | 0 | JUNE 1977 |
| 1-11 | 0 | MAY 1976 | 4-6 | 0 | JUNE 1977 |
| 1-12 | 0 | MAY 1976 | 4-7 | 0 | JUNE 1977 |
| 1-13 | 0 | JUNE 1977 | 4-8 | 0 | JUNE 1977 |
| 1-14 | 0 | JUNE 1977 | 4-9 | 0 | JUNE 1977 |
| 2-1 | 0 | JUNE 1977 | 4-10 | 0 | JUNE 1977 |
| 2-2 | 0 | JUNE 1977 | 4-11 | 0 | JUNE 1977 |
| 2-3 | 0 | JUNE 1977 | 4-12 | 0 | JUNE 1977 |
| 2-4 | 0 | JUNE 1977 | 4-13 | 0 | JUNE 1977 |
| 2-5 | 0 | JUNE 1977 | 4-14 | 0 | JUNE 1977 |
| 2-6 | 0 | JUNE 1977 | 4-15 | 0 | JUNE 1977 |
| 2-7 | 0 | JUNE 1977 | 4-16 | 0 | JUNE 1977 |
| 2-8 | 0 | JUNE 1977 | 4-16.1 | 0 | JUNE 1977 |
| 2-9 | 0 | JUNE 1977 | 4-17 | 0 | JUNE 1977 |
| 2-10 | 0 | JUNE 1977 | 4-18 | 0 | JUNE 1977 |
| 2-11 | 0 | JUNE 1977 | 4-19 | 0 | JUNE 1977 |
| 2-12 | 0 | JUNE 1977 | 4-20 | 0 | JUNE 1977 |
| 2-13 | 0 | JUNE 1977 | 4-21 | 0 | JUNE 1977 |
| 2-14 | 0 | JUNE 1977 | 4-22 | 0 | JUNE 1977 |
| 2-15 | 0 | JUNE 1977 | 4-23 | 0 | JUNE 1977 |
| 2-16 | 0 | JUNE 1977 | 4-24 | 0 | JUNE 1977 |
| 2-17 | 0 | JUNE 1977 | 4-25 | 0 | JUNE 1977 |
| 2-18 | 0 | JUNE 1977 | 4-26 | 0 | JUNE 1977 |
| 2-19 | 0 | JUNE 1977 | 4-27 | 0 | JUNE 1977 |
| 2-20 | 0 | JUNE 1977 | 4-28 | 0 | JUNE 1977 |
| 2-21 | 0 | JUNE 1977 | 5-1 | 0 | JUNE 1977 |
| 3-1 | 0 | JUNE 1977 | 5-2 | 0 | JUNE 1977 |
| 3-2 | 0 | JUNE 1977 | 5-3 | 0 | JUNE 1977 |

JUNE 1977

PAGE R-3

*** LIST OF EFFECTIVE PAGES ***

| PAGE | REV | DATE | PAGE | REV | DATE |
|--------|-----|-----------|---------|-----|-----------|
| 5-4 | 0 | JUNE 1977 | 8-4 | 0 | AUG 1975 |
| 5-5 | 0 | JUNE 1977 | 8-5 | 0 | JUNE 1977 |
| 5-6 | 0 | JUNE 1977 | 8-6 | 0 | DEC 1976 |
| 5-7 | 0 | JUNE 1977 | 8-7 | 0 | DEC 1976 |
| 5-8 | 0 | JUNE 1977 | 8-8 | 0 | JUNE 1977 |
| 5-9 | 0 | JUNE 1977 | 8-9 | 0 | JUNE 1977 |
| 5-10 | 0 | JUNE 1977 | 8-10 | 0 | JUNE 1977 |
| 5-11 | 0 | JUNE 1977 | 9-1 | 0 | JUNE 1977 |
| 5-12 | 0 | JUNE 1977 | 9-2 | 0 | JUNE 1977 |
| 5-13 | 0 | JUNE 1977 | 9-3 | 0 | JUNE 1977 |
| 5-14 | 0 | JUNE 1977 | 9-4 | 0 | JUNE 1977 |
| 5-15 | 0 | JUNE 1977 | 9-5 | 0 | JUNE 1977 |
| 5-16 | 0 | JUNE 1977 | 9-6 | 0 | JUNE 1977 |
| 5-17 | 0 | JUNE 1977 | 9-7 | 0 | JUNE 1977 |
| 6-1 | 0 | JUNE 1977 | 9-8 | 0 | JUNE 1977 |
| 6-2 | 0 | JUNE 1977 | 9-9 | 0 | JUNE 1977 |
| 6-3 | 0 | JUNE 1977 | 9-10 | 0 | JUNE 1977 |
| 6-4 | 0 | JUNE 1977 | 9-11 | 0 | JUNE 1977 |
| 6-5 | 0 | JUNE 1977 | 9-12 | 0 | JUNE 1977 |
| 6-6 | 0 | JUNE 1977 | 9-13 | 0 | JUNE 1977 |
| 6-7 | 0 | JUNE 1977 | 9-14 | 0 | JUNE 1977 |
| 6-8 | 0 | JUNE 1977 | 9-15 | 0 | JUNE 1977 |
| 6-9 | 0 | JUNE 1977 | 9-16 | 0 | JUNE 1977 |
| 7-1 | 0 | JUNE 1975 | 9-17 | 0 | JUNE 1977 |
| 7-2 | 0 | JUNE 1977 | 9-18 | 0 | JUNE 1977 |
| 7-3 | 0 | JUNE 1977 | 9-19 | 0 | JUNE 1977 |
| 7-4 | 0 | JUNE 1977 | 9-20 | 0 | JUNE 1977 |
| 7-5 | 0 | JUNE 1977 | 9-21 | 0 | JUNE 1977 |
| 7-6 | 0 | JUNE 1977 | 10-1 | 0 | JUNE 1977 |
| 7-7 | 0 | JUNE 1977 | 10-2 | 0 | JUNE 1977 |
| 7-8 | 0 | JAN 1973 | 10-3 | 0 | JUNE 1977 |
| 7-9 | 0 | JUNE 1977 | 10-4 | 0 | JUNE 1977 |
| 7-10 | 0 | JUNE 1977 | 10-5 | 0 | JUNE 1977 |
| 7-11 | 0 | JUNE 1977 | 10-6 | 0 | JUNE 1977 |
| 7-12 | 0 | JUNE 1977 | 10-7 | 0 | JUNE 1977 |
| 7-13 | 0 | JUNE 1977 | 10-8 | 0 | JUNE 1977 |
| 7-14 | 0 | JUNE 1977 | 10-9 | 0 | JUNE 1977 |
| 7-15 | 0 | JUNE 1977 | 10-10 | 0 | JUNE 1977 |
| 7-16 | 0 | JUNE 1977 | 10-11 | 0 | JUNE 1977 |
| 7-17 | 0 | JUNE 1977 | 10-12 | 0 | JUNE 1977 |
| 7-18 | 0 | JUNE 1977 | 10-13 | 0 | JUNE 1977 |
| 7-19 | 0 | JUNE 1977 | 10-14 | 0 | JUNE 1977 |
| 7-20 | 0 | JUNE 1977 | 10-15 | 0 | JUNE 1977 |
| 7-21 | 0 | JUNE 1977 | 10-16 | 0 | JUNE 1977 |
| 7-22 | 0 | JUNE 1977 | 10-17 | 0 | JUNE 1977 |
| 7-22.1 | 0 | JUNE 1977 | 10-18 | 0 | JUNE 1977 |
| 7-22.2 | 0 | JUNE 1977 | 10-19 | 0 | JUNE 1977 |
| 8-1 | 0 | JUNE 1977 | 10-20 | 0 | JUNE 1977 |
| 8-2 | 0 | JUNE 1977 | 10-20.1 | 0 | JUNE 1977 |
| 8-3 | 0 | JUNE 1977 | 10-21 | 0 | JUNE 1977 |

JUNE 1977

PAGE R-4

*** LIST OF EFFECTIVE PAGES ***

| PAGE | REV | DATE | PAGE | REV | DATE |
|-------|-----|-----------|-------|-----|-----------|
| 10-22 | 0 | JUNE 1977 | 15-4 | 0 | JUNE 1977 |
| 11-1 | 0 | JUNE 1977 | 15-5 | 0 | JUNE 1977 |
| 11-2 | 0 | JUNE 1977 | 15-6 | 0 | JUNE 1977 |
| 11-3 | 0 | JUNE 1977 | 15-7 | 0 | JUNE 1977 |
| 11-4 | 0 | JUNE 1977 | 15-8 | 0 | JUNE 1977 |
| 11-5 | 0 | JUNE 1977 | 15-9 | 0 | JUNE 1977 |
| 11-6 | 0 | JUNE 1977 | 15-10 | 0 | JUNE 1977 |
| 11-7 | 0 | JUNE 1977 | 15-11 | 0 | JUNE 1977 |
| 11-8 | 0 | JUNE 1977 | 15-12 | 0 | JUNE 1977 |
| 11-9 | 0 | JUNE 1977 | 15-13 | 0 | JUNE 1977 |
| 11-10 | 0 | JUNE 1977 | 15-14 | 0 | JUNE 1977 |
| 11-11 | 0 | JUNE 1977 | 15-15 | 0 | JUNE 1977 |
| 11-12 | 0 | JUNE 1977 | 15-16 | 0 | JUNE 1977 |
| 11-13 | 0 | JUNE 1977 | 15-17 | 0 | JUNE 1977 |
| 11-14 | 0 | JUNE 1977 | 15-18 | 0 | JUNE 1977 |
| 11-15 | 0 | JUNE 1977 | 15-19 | 0 | JUNE 1977 |
| 11-16 | 0 | JUNE 1977 | 15-20 | 0 | JUNE 1977 |
| 11-17 | 0 | JUNE 1977 | 15-21 | 0 | JUNE 1977 |
| 11-18 | 0 | JUNE 1977 | 15-22 | 0 | JUNE 1977 |
| 11-19 | 0 | JUNE 1977 | 15-23 | 0 | JUNE 1977 |
| 11-20 | 0 | JUNE 1977 | 15-24 | 0 | JUNE 1977 |
| 12-1 | 0 | JUNE 1977 | 15-25 | 0 | JUNE 1977 |
| 12-2 | 0 | SEPT 1975 | 15-26 | 0 | JUNE 1977 |
| 12-3 | 0 | JUNE 1977 | | | |
| 12-4 | 0 | JUNE 1977 | | | |
| 12-5 | 0 | JUNE 1977 | | | |
| 12-6 | 0 | JUNE 1977 | | | |
| 12-7 | 0 | JUNE 1977 | | | |
| 12-8 | 0 | JUNE 1977 | | | |
| 12-9 | 0 | JUNE 1977 | | | |
| 12-10 | 0 | JUNE 1977 | | | |
| 12-11 | 0 | JUNE 1977 | | | |
| 12-12 | 0 | JUNE 1977 | | | |
| 12-13 | 0 | JUNE 1977 | | | |
| 12-14 | 0 | JUNE 1977 | | | |
| 12-15 | 0 | JUNE 1977 | | | |
| 12-16 | 0 | JUNE 1977 | | | |
| 12-17 | 0 | JUNE 1977 | | | |
| 13-1 | 0 | JUNE 1977 | | | |
| 13-2 | 0 | JUNE 1977 | | | |
| 13-3 | 0 | JUNE 1977 | | | |
| 13-4 | 0 | JUNE 1977 | | | |
| 13-5 | 0 | JUNE 1977 | | | |
| 14-1 | 0 | JUNE 1977 | | | |
| 14-2 | 0 | JUNE 1977 | | | |
| 14-3 | 0 | JUNE 1977 | | | |
| 14-4 | 0 | JUNE 1977 | | | |
| 15-1 | 0 | JUNE 1977 | | | |
| 15-2 | 0 | JUNE 1977 | | | |
| 15-3 | 0 | JUNE 1977 | | | |

*** ABBREVIATIONS ***

THROUGHOUT THIS MANUAL, THE FOLLOWING ABBREVIATIONS WILL BE USED TO INDICATE REFERENCE MANUALS:

| ABBREVIATION | MANUAL | FORM NO. |
|--------------|---|-----------------------------|
| CCIRM | COMPUTER CENTER INTRODUCTORY REF MANUAL | CMLD-77-10 |
| CCLIB | COMPUTER CENTER LIBRARIES | CMLD-77-12 |
| CCLIB/N | CCLIB/NSRDC (SUBPROGRAMS) | CMLD-77-15 |
| CCLIB/P | CCLIB/PROFIL (PROCEDURES) | CMLD-77-16 |
| CCLIB/U | CCLIB/UTILITY (PROGRAMS) | CMLD-77-17 |
| CCRM | COMPUTER CENTER REFERENCE MANUAL (THIS MANUAL) | CMLD-77-11 THRU REV 0 |
| CCURM | CYBER COMMON UTILITIES REFERENCE MANUAL | 60493300 B |
| COBOL | COBOL VERSION 4 REFERENCE MANUAL | OR 60495600 D 60384100 F |
| CONV | 3.3 TO 3.4 CONVERSION AIDS | OR 60496800 B 60358200 C |
| DEPUG | FORTRAN EXTENDED DEBUG USER'S GUIDE | 60329400 B |
| FORM | FORM REFERENCE MANUAL | 60307000 F |
| FTN | FORTRAN EXTENDED V4 REFERENCE MANUAL (NOS/BE) | 60497800 B |
| FTN/S | FORTRAN EXTENDED V4 REFERENCE MANUAL (SCOPE) | 60305600 K |
| INT | INTERCOM REFERENCE MANUAL | 60307100 F |
| INTCCB | INTERCOM FOR COBOL USERS | OR 60494600 C |
| INTFTN | INTERCOM FOR FORTRAN USERS | 60385700 B 60359700 B |
| LOADER | LOADER REFERENCE MANUAL (NOS/BE) | 60429800 C |
| LOADER/S | LOADER REFERENCE MANUAL (SCOPE) | 60344200 G |
| NOSBE | NOS/BE 1 REFERENCE MANUAL | 60493800 C |
| POLICY | COMPUTER CENTER POLICY | |
| RM | RECORD MANAGER REFERENCE MANUAL | 60307300 J |
| RMCCB | RECORD MANAGER FOR COBOL USERS | 60385300 A |
| RMFCR | RECORD MANAGER FOR FORTRAN USERS | 60385200 B |
| RMUG | RECORD MANAGER USER'S GUIDE | 60359600 C |
| SCOPE | SCOPE 3.4 REFERENCE MANUAL | 60307200 L |
| SORT | SORT/MERGE REFERENCE MANUAL | 60343900 J |
| SUG | SCOPE 3.4 USER'S GUIDE | 60358700 A |
| UPDATE | UPDATE REFERENCE MANUAL | 60342500 G |

MOST OF THESE MANUALS ARE IN THE REFERENCE CENTERS. (SEE CCRM, 13-2 THRU 13-4)

*** PAGE NUMBERING SCHEME ***

EACH CHAPTER IN THIS MANUAL IS NUMBERED CONSECUTIVELY STARTING WITH PAGE 1, EXCEPT AS FOLLOWS:

- A) CERTAIN PAGES WHICH DESCRIBE SUCH THINGS AS PROGRAMS IN A LIBRARY; SPECIFIC SUBROUTINES OR PROGRAMS WHICH ARE SOMEHOW RELATED; AVAILABLE CATALOGUED PROCEDURES; ETC., MAY EXPAND AS REVISIONS ARE ISSUED AND SO ARE GIVEN ONE PAGE NUMBER (EG, 7-22). IF THE DESCRIPTION REQUIRES MORE THAN ONE PAGE, THESE PAGES ARE NUMBERED .1, .2, ETC. (EG, 7-22.1, 7-22.2, ...). REFERENCES IN THE MANUAL ARE TO THE FIRST PAGE (E.G., 7-22) AND THE SPECIFIC ROUTINE REFERENCED CAN THEN BE FOUND IN ALPHABETICAL ORDER. THE SPECIFIC PAGES INVOLVED ARE:

| | |
|-------|--|
| 4-16 | FTN-CALLABLE UTILITIES |
| 7-22 | HOW TO USE SOME PROCEDURES IN LIBRARY PROFIL |
| 10-20 | HOW TO USE SOME PROGRAMS IN LIBRARY UTILITY |

- B) AFTER PUBLICATION, AS ADDITIONAL PAGES ARE ADDED, THOSE WHICH ARE ADDED TO THE END OF A CHAPTER WILL CONTINUE THE NUMBERING SCHEME; THOSE INSERTED IN THE MIDDLE OF A CHAPTER WILL HAVE .1, .2, ETC., ADDED TO THE PAGE NUMBER THEY ARE TO FOLLOW (EG, 7-6.1 WILL FOLLOW 7-6).

*** REVISIONS ***

THIS PRINTING IS A MAJOR REVISION OF THE SEPTEMBER 1975 COMPUTER CENTER REFERENCE MANUAL (CMD-24-75). IT INCLUDES INFORMATION ON NEW HARDWARE AND REFLECTS THOSE CHANGES AND ADDITIONS RESULTING FROM THE INSTALLATION OF THE NOS/BE 1.0 OPERATING SYSTEM IN FEBRUARY 1977. ALL PAGES WITH A DATE (AT THE TOP OF THE PAGE) AFTER JAN 1977 ARE EITHER NEW OR REVISED.

ADDITIONAL OR CORRECTION PAGES WILL BE ISSUED AT INTERVALS.

TABLE OF CONTENTS

| | PAGE |
|--|------|
| PREFACE | |
| REVISION RECORD | R-1 |
| LIST OF EFFECTIVE PAGES | R-2 |
| ABBREVIATIONS | I |
| PAGE NUMBERING SCHEME | II |
| REVISIONS | II |
| 1 INTRODUCTION TO CDC 6000 COMPUTERS | |
| HARDWARE CONFIGURATION | 1-1 |
| NOS/BE 1.0 | 1-3 |
| JOB PROCESSING | 1-4 |
| PRIVACY ACT OF 1974 | 1-5 |
| KEYPUNCH CHARACTERS | 1-6 |
| USE OF ALTERNATE KEYPUNCH CHARACTERS | 1-6 |
| INTERNAL DATA STRUCTURE | 1-7 |
| DTNSRDC CHARACTER SET | 1-9 |
| OCTAL-DECIMAL CONVERSION | 1-11 |
| CDC 6000 REAL FLOATING | 1-11 |
| OCTAL-DECIMAL FRACTIONS | 1-12 |
| CONTROL CARD FIELD LENGTHS | 1-13 |
| RECOMMENDED FIELD LENGTHS | 1-14 |
| 2 JOB PROCESSING CONTROL CARDS | |
| INTRODUCTION | 2-1 |
| CONTROL CARD FORMATS | 2-2 |
| JOB CARD | 2-3 |
| CHARGE CARD | 2-4 |
| PRIVATE | 2-5 |
| OTHER NOS/BE CONTROL CARDS | 2-6 |
| DISPOSE | 2-8 |
| SPECIAL FORMS CODES | 2-9 |
| ROUTE | 2-10 |
| ROUTE EXAMPLES | 2-12 |
| LOADER | 2-13 |
| TYPES OF LOADING | 2-13 |
| LOADER CONTROL CARDS | 2-15 |
| LDSET | 2-16 |
| LOADER-RELATED CONTROL CARDS | 2-17 |
| SEGMENTATION | 2-19 |
| SEGMENTATION CAUTIONS | 2-20 |
| ERROR MESSAGES | 2-21 |
| LOADER AND NOS/BE ERRORS | 2-21 |
| MODE ERRORS | 2-21 |
| TAPE PARITY ERRORS | 2-21 |

PAGE

3 FILES AND TAPES

| | |
|--|------|
| INTRODUCTION | 3-1 |
| RECORD MANAGER | 3-1 |
| FILE ORGANIZER AND RECORD MANAGER (FORM) | 3-1 |
| CHARACTERISTICS OF MASS STORAGE DEVICES | 3-2 |
| PERMANENT FILES | 3-3 |
| AUTOMATIC PERMANENT FILE PURGE | 3-3 |
| USER AUDIT | 3-4 |
| PERMANENT FILE COMMANDS | 3-5 |
| HINTS ON PERMANENT FILE USAGE | 3-6 |
| PERMANENT FILE USER RETURN CODES | 3-7 |
| PERMANENT FILE EXAMPLES | 3-8 |
| DEVICE SETS | 3-10 |
| DEVICE SET CONTROL CARDS | 3-11 |
| DEVICE SET UTILITIES | 3-12 |
| TAPE CHARACTERISTICS | 3-13 |
| TAPE ASSIGNMENT | 3-14 |
| TAPE USAGE AND CLEANING | 3-14 |
| VSN | 3-15 |
| LABELLED TAPES | 3-16 |
| UNLABELLED TAPES | 3-17 |
| DESCRIBING A FILE | 3-18 |
| INDEXED SEQUENTIAL | 3-19 |

4 FORTRAN EXTENDED

| | |
|--|------|
| INTRODUCTION | 4-1 |
| FTN CONTROL CARD PARAMETERS | 4-1 |
| PROGRAM CARD | 4-3 |
| CONVERSION TO FORTRAN EXTENDED | 4-4 |
| PITFALLS IN FORTRAN | 4-5 |
| FORTRAN IV EXTENDED - VERSION 4 | 4-10 |
| TIME-SHARING FORTRAN | 4-13 |
| ERROR MESSAGES | 4-14 |
| FORTRAN COMPILATION ERRORS | 4-14 |
| FORTRAN OBJECT TIME ERRORS | 4-14 |
| READING DUMPS - FORTRAN DEBUGGING | 4-15 |
| FORTRAN CALLABLE UTILITIES | 4-16 |
| SAMPLE FORTRAN DECK SETUPS | 4-17 |
| SAMPLE FORTRAN INPUT/OUTPUT | 4-20 |
| OVERLAY | 4-21 |
| SEGMENTATION | 4-22 |
| SETTING UP OVERLAY | 4-23 |
| SAMPLE OVERLAY | 4-24 |
| COMPILE, LOAD AND CATALOG ABSOLUTE PROGRAM | 4-25 |
| MINNESOTA FORTRAN (MNF) | 4-26 |
| RATIONAL FORTRAN (RATFOR) | 4-28 |

5 COBOL

| | |
|--------------------------------------|------|
| INTRODUCTION TO COBOL 4 | 5-1 |
| COBOL CONTROL CARD PARAMETERS | 5-1 |
| COBOL COMPILATION ERRORS | 5-2 |
| CONVERSION TO COBOL FROM IBM S/360 | 5-3 |
| COBOL 4 VERSUS COBOL 3 | 5-7 |
| SAMPLE COBOL DECK SETUPS | 5-10 |
| OVERLAY FOR COBOL | 5-12 |
| COMPILE WITH SUBROUTINES AND OVERLAY | 5-13 |
| COBOL 5 | 5-14 |
| CONTROL CARD PARAMETERS | 5-15 |
| COBOL 5 VERSUS COBOL 4 | 5-16 |
| COBOL 4 TO COBOL 5 CONVERSION AID | 5-17 |

6 UTILITIES

| | |
|-----------------------------|-----|
| NOS/BE UTILITIES | 6-1 |
| COPYCF/COPYBF/COPYCR/COPYBR | 6-1 |
| SKIPF/SKIPB | 6-1 |
| COPYL/COPYLM | 6-2 |
| ITEMIZE | 6-2 |
| COPYN | 6-3 |
| BKSP/COMBINE/COMPARE/COPY | 6-3 |
| OTHER UTILITIES | 6-4 |
| COPYE | 6-4 |
| COPYF/COPYR | 6-5 |
| COPYRM | 6-6 |
| COPYSF(COPYSBF)/COPYSR | 6-6 |
| SAMPLE UTILITY SETUPS | 6-7 |

7 LIBRARY MAINTENANCE UTILITIES

| | |
|--------------------------------------|------|
| UPDATE | 7-2 |
| UPDATE CARD PARAMETERS | 7-2 |
| UPDATE DIRECTIVES | 7-3 |
| SAMPLE UPDATE SETUPS | 7-4 |
| EDITLIB | 7-9 |
| EDITLIB CONTROL CARD | 7-9 |
| EDITLIB DIRECTIVES | 7-9 |
| EDITLIB EXAMPLES | 7-12 |
| FTN OVERLAY IN A LIBRARY | 7-15 |
| CATALOGUED PROCEDURES (BEGIN/REVERT) | 7-18 |
| DUPLICATE CONTROL CARDS (DUP/ENDUP) | 7-19 |
| RANDOM PROCEDURE FILES | 7-20 |
| PUBLIC-ACCESS PROCEDURES - PROFIL | 7-21 |

8 INTERACTIVE GRAPHICS

| | |
|----------------------------------|-----|
| INTRODUCTION TO IGS | 8-1 |
| DTNSRDC GRAPHICS | 8-2 |
| IGS EDITLIB EXAMPLES | 8-3 |
| IGS UTILITIES | 8-4 |
| SIGNON | 8-6 |
| TEKTRONIX 4015 | 8-8 |
| TEKTRONIX 4662 PLOTTER/DIGITIZER | 8-9 |

9 INTERCOM 4

| | |
|--------------------------------|------|
| SIGN-ON PROCEDURES | 9-1 |
| USER TURNKEY PASSWORD | 9-2 |
| CORRECTING AND INTERRUPTING | 9-3 |
| INTERCOM USER LIMITATIONS | 9-4 |
| PAPER TAPE USE | 9-5 |
| PROGRAM CONSTRUCTION - EDITOR | 9-6 |
| BATCH RUNS FROM TELETYPE | 9-8 |
| LEAVING AND RE-ENTERING EDITOR | 9-9 |
| INTERCOM BACKUP DECKS | 9-9 |
| SIGN-OFF PROCEDURES | 9-9 |
| SYSTEM BULLETIN | 9-9 |
| LOCATING JOBS | 9-10 |
| Q COMMAND | 9-10 |
| QS COMMAND | 9-11 |
| MESSAGES BETWEEN CONSOLES | 9-12 |
| SAVING FILES | 9-14 |
| STORE, FETCH, DISCARD COMMANDS | 9-14 |
| ADDITIONAL INTERCOM COMMANDS | 9-15 |
| XEQ COMMAND | 9-16 |
| XEQ EXAMPLES | 9-17 |
| PAGE | 9-18 |
| TEXT EDITOR (NETED) | 9-21 |

10 LIBRARY AND OTHER ROUTINES

| | |
|-----------------------------|------|
| NOS/BE 1.0 SYSTEM LIBRARIES | 10-1 |
| LIBRARIES OF SUBPROGRAMS | 10-2 |
| SYSMISC | 10-2 |
| ARLNALG | 10-3 |
| EDSTAT | 10-3 |
| EISPACK | 10-3 |
| FUNPAK | 10-3 |
| IMSL | 10-4 |
| MSL | 10-4 |
| NSRDC | 10-5 |
| SSP | 10-6 |
| PLOTTING LIBRARIES | 10-7 |
| PRINTER PLOTS | 10-7 |
| OFF-LINE PLOTTERS | 10-7 |

| | PAGE |
|------------------------------------|-------|
| CALCOMP PLOTTERS | 10-7 |
| CALCOMP 936 PEN PLOTTER | 10-8 |
| 1700 CALCOMP PEN PLOTTER | 10-8 |
| CALCOMP 763 PEN PLOTTER | 10-8 |
| CALCOMP FUNCTIONAL PACKAGE | 10-9 |
| EXAMPLES | 10-10 |
| SC4020 TO CALCOMP | 10-11 |
| CALCOMP THREE-D SOFTWARE | 10-12 |
| SC-4020 MICROFILM PLOTTER ROUTINES | 10-13 |
| SC-4020 - USER INFORMATION | 10-13 |
| SAMPLE 4020 USAGE | 10-15 |
| SC4060 - META USER INFORMATION | 10-16 |
| DISSPLA | 10-17 |
| LIBRARIES OF MAIN PROGRAMS | 10-19 |
| BIMED | 10-19 |
| BIMEDP | 10-19 |
| UTILITY | 10-20 |
| MISCELLANEOUS PROGRAMS | 10-21 |
| VIM PROGRAMMING PACKAGES | 10-22 |
| OTHER SOURCES OF PROGRAMS | 10-23 |
| PROGRAM DOCUMENTATION | 10-24 |
| SOURCE CODE | 10-24 |

11 SPECIAL PURPOSE PROGRAMMING LANGUAGES

| | |
|--------------------------------------|-------|
| ALGOL | 11-1 |
| AUTOMATICALLY PROGRAMMED TOOLS (APT) | 11-2 |
| CHECKPOINT/RESTART | 11-3 |
| OPERATOR CHECKPOINT | 11-5 |
| COMRADE | 11-6 |
| COMPASS | 11-6 |
| DATA MANAGEMENT SYSTEM 170 (DMS170) | 11-7 |
| MARS VI | 11-8 |
| NASTRAN | 11-11 |
| MIMIC | 11-12 |
| PERT | 11-12 |
| QWICK QWERY | 11-12 |
| RIQS | 11-12 |
| SHARP | 11-13 |
| SORT/MERGE | 11-14 |
| GPSS | 11-17 |
| OMNITAR | 11-17 |
| PL/I SUBSET | 11-17 |
| SCORE | 11-17 |
| SPSS | 11-18 |
| SIMSCRIPT I.5 | 11-19 |
| SIMSCRIPT II.5 | 11-19 |
| SNOROL | 11-19 |
| SYSTEM 2000 | 11-19 |
| VENUS | 11-20 |

PAGE

12 OTHER COMPUTER CENTER EQUIPMENT

| | |
|---|-------|
| 1700 COMPUTER TERMINAL | 12-1 |
| 1700 REFERENCE MANUALS | 12-2 |
| 200 USER TERMINAL | 12-3 |
| NOVA 800 | 12-3 |
| REMOTE BATCH PROCESSING - MEDIUM SPEED | 12-4 |
| PRINTER | 12-4 |
| CARD READER | 12-5 |
| PRINT AND CARD READER | 12-5 |
| QUEUE AND STATUS DISPLAYS | 12-5 |
| MANIPULATING QUEUE-RESIDENT FILES | 12-6 |
| SOME ERROR MESSAGES | 12-7 |
| STROMBERG DATAGRAPHIX (SC4060) | 12-8 |
| COM - DATAGRAPHS 4530 | 12-9 |
| CALCOMP MODEL 936 | 12-10 |
| CALCOMP MODEL 763 | 12-11 |
| CALCOMP PLOT REQUEST | 12-11 |
| 763 CALCOMP PAPER AND GRID COLOR | 12-12 |
| GSA-ATLANTA TIME SHAPING SYSTEM | 12-13 |
| AUTOMATED DRAFTING AND GRAPHIC DATA REDUCTION | |
| STATION | 12-14 |
| ALTEK AC-90 DIGITIZING SYSTEM | 12-14 |
| GRAPHICS DATA REDUCTION AND DESIGN STATION | 12-14 |
| REMOTE JOB ENTRY TO TI-ASC | 12-15 |
| NAVY LABORATORY COMPUTER NETWORK (NALCON) | 12-16 |
| FARRINGTON OCR | 12-17 |

13 USER INTERFACE

| | |
|---------------------------------|------|
| GENERAL INFORMATION | 13-1 |
| USER TROUBLE FORMS | 13-1 |
| GRIPES | 13-1 |
| COMPUTER REFERENCE CENTERS | 13-2 |
| COMPUTER REFERENCE MANUALS | 13-3 |
| CENTRAL SITE EAM PROCESSING | 13-5 |
| ADP CONTROL CENTER | 13-5 |
| PUNCHED CARDS FROM CENTRAL SITE | 13-5 |

JUNE 1977

PAGE IX

PAGE

14 USER HELPS

| | |
|---|------|
| JOB RERUN | 14-1 |
| ABORT PROCESSING | 14-1 |
| CP TIME LIMIT | 14-1 |
| IO TIME LIMIT | 14-1 |
| MASS STORAGE LIMIT | 14-1 |
| MESSAGE LIMIT EXCEEDED | 14-1 |
| REPRIEVE | 14-1 |
| CONVERSION AIDS | 14-2 |
| COMMUNICATIONS WITH OPERATOR | 14-3 |
| PM - CARRIAGE CONTROL DISPOSE AT REMOTE | 14-3 |
| SPECIAL CARRIAGE CONTROL | 14-4 |
| DAYFILE MESSAGE DISPLAY | 14-4 |
| RESERVED WORDS | 14-4 |

15 INDEX

***** INTRODUCTION TO CDC 6000 COMPUTERS *****

THE COMPUTATION, MATHEMATICS AND LOGISTICS DEPARTMENT HAS THREE CONTROL DATA 6000 COMPUTERS. THE CDC 6400 IS LOCATED IN BUILDING 191; THE CDC 6600 AND CDC 6700 ARE LOCATED IN BUILDING 17. THE 6600 AND 6700 OPERATE AS A SINGLE SYSTEM SHARING PERMANENT FILES AND LOAD LEVELING JOBS. THESE LARGE DIGITAL COMPUTERS CAN RUN INDEPENDENTLY.

THE 6400 AND 6600 EACH HAVE A SINGLE CENTRAL PROCESSING UNIT (CPU). THE 6700 IS A DUAL PROCESSOR, WITH TWO CPU'S (6600 AND 6400) WHICH OPERATE CONCURRENTLY ON DIFFERENT JOBS BOTH STORED IN CENTRAL MEMORY (CM). THE CM ON EACH SYSTEM IS 400000 OCTAL (OR 131,072 DECIMAL) 60-BIT WORDS.

PERIPHERAL PROCESSORS (PP'S) ARE SMALL COMPUTERS (4096 MEMORY OF 12-BIT WORDS) WHICH HANDLE ALL INPUT AND OUTPUT (I/O). THERE ARE 10 PP'S ON THE 6400; 20 EACH ON THE 6600 AND 6700.

THERE ARE 12 I/O CHANNELS ON THE 6400; 24 EACH ON THE 6600 AND 6700. ALL PERIPHERAL EQUIPMENT, INCLUDING PRINTERS, TAPE AND DISK DRIVES, AND MULTIPLEXORS FOR REMOTE TERMINALS, INTERFACES WITH THE CENTRAL SYSTEM THROUGH THE PP'S VIA THE I/O CHANNELS.

*** HARDWARE CONFIGURATION ***

CDC 6400 (MAINFRAME C - MFC)

131,072 60-BIT WORD MEMORY
10 PERIPHERAL PROCESSORS
MODEL 844 DISK DRIVES (8 SPINDLES)
4 SEVEN-TRACK TAPE DRIVES
2 PORTS FOR 1700 TERMINALS
2 TELEPHONE LINES FOR BCD 2000-BAUD TERMINALS (227-1950)
1 TELEPHONE LINE FOR ASCII 2000-BAUD TERMINALS (227-2820)
2 TELEPHONE LINES FOR ASCII 4800-BAUD TERMINALS (227-1674)
20 TELEPHONE LINES FOR 300-BAUD INTERACTIVE TERMINALS
(229-7700) 5
(229-0980) 15

JUNE 1977

PAGE 1-2

CDC 6600 (MAINFRAME B - MFB)
CDC 6700 (MAINFRAME A - MFA)

DUAL CENTRAL PROCESSORS ON 6700 (CPA=6600, CPB=6400)
131,072 60-BIT WORD MEMORY (EACH)
20 PERIPHERAL PROCESSORS (EACH)
MODEL 6638 NON-REMOVABLE DISK (EACH)
MODEL 844 DISK DRIVES (1 SPINDLE EACH + 26 SHARED)
SIX 7-TRACK TAPE DRIVES (6600:2, 6700:4)
TWO 9-TRACK TAPE DRIVES (6700)
3 1700 COMPUTERS WITH 274 CONSOLES FOR IGS (6700)
1 1700 TERMINAL (6700)
1 2550 DATA CONCENTRATOR (6600)
9 TELEPHONE LINES FOR ASCII/BCD 4800-BAUD TERMINALS
(227-1000)
5 TELEPHONE LINES FOR ASCII/BCD 2000-BAUD TERMINALS
(227-1958) 3
(227-1970) 2
2 TELEPHONE LINES FOR 1200-BAUD INTERACTIVE TERMINALS
BELL-212 (227-2172) 1
BELL-212 (227-2178) 1
VADIC (FUTURE)
32 TELEPHONE LINES FOR 300-/110-BAUD INTERACTIVE TERMINALS
(229-6000) 20
(229-6610) 12
1 ANNAPOLIS TELEPHONE LINE FOR 300-BAUD INTERACTIVE TERMINALS
(267-3206)
1 ANNAPOLIS TELEPHONE LINE FOR 110-BAUD INTERACTIVE TERMINALS
(267-3160)
ADDITIONAL LINES ON 6600 ARE:
2 TELEPHONE LINES FOR BCD 4800-BAUD TERMINALS (227-2167)
(227-2864)
2 TELEPHONE LINES FOR ASCII 2000-BAUD TERMINALS (227-2794)
4 TELEPHONE LINES FOR BCD 2000-BAUD TERMINALS (227-2772)
10 TELEPHONE LINES FOR ASCII 4800-BAUD TERMINALS (227-2760) 3
(227-1997) 3
(227-2883) 4

THESE SYSTEMS ARE COMPATIBLE AND BINARY PROGRAMS GENERATED ON ONE SYSTEM MAY BE RUN ON ANOTHER. HOWEVER, THE 6400 PERMANENT FILE SYSTEM IS COMPLETELY SEPARATE FROM THE 6600/6700, HENCE USERS SHOULD CHOOSE THE 6400 OR THE 6600/6700 FOR PROCESSING A CLASS OF PROGRAMS TO AVOID MAINTAINING AND PAYING FOR DUPLICATE SETS OF PERMANENT FILES.

*** NOS/BE 1.0 ***

THE OPERATING SYSTEM FOR ALL CDC 6000 COMPUTERS AT DTNSRDC IS CALLED THE NETWORK OPERATING SYSTEM/BATCH ENVIRONMENT, VERSION 1.0, (NOS/BE 1.0) AND DIFFERS ONLY SLIGHTLY FROM THE STANDARD NOS/BE SYSTEM OF CDC. THE INTERACTIVE SYSTEM FOR TELETYPE-COMPATIBLE TERMINALS (6600 AND 6400 ONLY), MEDIUM SPEED BATCH AND HIGH SPEED REMOTE TERMINALS IS CALLED INTERCOM. THE INTERACTIVE GRAPHICS SYSTEM (IGS) RUNS FROM THOSE 1700 TERMINALS WITH 274 CONSOLES. THE UNIT RECORD EQUIPMENT (CARD READERS, LINE PRINTERS, ETC.) AT THE CENTRAL SITE IS CONTROLLED BY A SUBSYSTEM CALLED JANUS, WHICH OCCUPIES ONE CONTROL POINT. THE REMAINING CONTROL POINTS ARE AVAILABLE FOR INTERACTIVE, USER BATCH AND OTHER SYSTEM JOBS.

EACH CPU HAS 24 REGISTERS FOR INFORMATION ON WHICH TO OPERATE: 8 ADDRESS (A) REGISTERS, 8 OPERAND (X) REGISTERS AND 8 INCREMENT (B) REGISTERS. THE 6600 CPU CAN PERFORM SEVERAL DIFFERENT FUNCTIONS SIMULTANEOUSLY (2 ADDS, 2 MULTIPLIES, 2 INCREMENTS, 1 DIVIDE, 1 SHIFT, 1 BCCLEAN, 1 BRANCH) AND IT HAS A BUFFER OF 8 CM WORDS OF INSTRUCTIONS, CALLED AN INSTRUCTION STACK. THESE TWO FEATURES ALLOW THE 6600 TO EXECUTE FASTER THAN THE 6400 CPU. ON THE 6700, ONE JOB MAY BE EXECUTED PARTLY ON THE 6600 CPU AND PARTLY ON THE 6400 CPU AT DIFFERENT TIMES, BECAUSE THE CPU'S ARE SHARED BY MANY JOBS.

PERMANENT FILES (USER PROGRAM AND DATA FILES RETAINED FOR FREQUENT USE) RESIDE ON THE 844 DISK DRIVES. MOST OF THE OPERATING SYSTEM RESIDES ON MASS STORAGE ON 844 SYSTEM DISK. USER FILES, IF NOT SPECIFICALLY REQUESTED ON A TAPE OR USER DEVICE SET, WILL BE ASSIGNED TO AVAILABLE AREAS OF MASS STORAGE.

PRIOR TO APRIL 1975 THE PRINCIPAL OPERATING SYSTEM WAS SCOPE 3.3. SOURCE PROGRAMS FROM SCOPE 3.3 WERE RECOMPILED UNDER SCOPE 3.4 OR NOS/BE 1.0. OBJECT DECKS MAY NOT BE MIXED. (SEE CCRM, 4-12: ITEM 16.)

*** JOB PROCESSING ***

A JOB ENTERING THE SYSTEM GOES TO THE INPUT QUEUE AND WAITS FOR THE SYSTEM TO CHECK THAT THE RESOURCES REQUIRED BY THE JOB (E.G., CM AND TAPE DRIVES) ARE AVAILABLE. WHEN A JOB HAS THE HIGHEST PRIORITY (PRIORITY INCREASES AS A JOB WAITS) AND RESOURCES ARE AVAILABLE, THE JOB IS ASSIGNED A JOB DESCRIPTOR (JDT) ENTRY AND MAY OCCUPY ONE OF THE CONTROL POINTS FOR ACTIVE JOBS (UP TO 15 CONTROL POINTS). THE JOB IS LOADED INTO AN ASSIGNED AREA OF CM CALLED THE FIELD LENGTH (FL). THE JOB MAY NOT ACCESS ANY AREA OF MEMORY OUTSIDE ITS OWN FL, BUT THE SYSTEM WILL VARY THE SIZE OF FL DURING EXECUTION. A USER MAY VARY THE MAXIMUM FL BY THE CONTROL CARD RFL (CCRM, 2-18). A JOB MAY BE AUTOMATICALLY SWAPPED OUT TO ALLOW OTHER JOBS TO RUN WHILE WAITING FOR RESOURCES SUCH AS FL, TAPE MOUNT, OR CENTRAL PROCESSOR (CP) TIME. THE JOB WILL SHARE THE CPU(S) AND THE PP'S WITH THE OTHER ACTIVE JOBS UNTIL IT IS COMPLETED. THEN THE RESULTS OF THE JOB WAIT IN THE OUTPUT QUEUE FOR PRINTING AND/OR PUNCHING.

THE USER CONTROLS HIS JOB'S EXECUTION THROUGH CONTROL CARDS WHICH MUST APPEAR IN THE FIRST LOGICAL RECORD OF HIS PROGRAM DECK. ONE CARD IS PROCESSED AT A TIME AND ANY CONTROL CARDS REQUIRING DATA (SUCH AS A FORTRAN COMPILATION) WILL TAKE THE NEXT LOGICAL RECORD IN THE JOB'S INPUT STREAM.

A RECORD OF THE ACTIVITY ON THE SYSTEM IS RECORDED IN THE DAYFILE. THE LAST PAGE OF OUTPUT FOR EVERY JOB CONTAINS THAT JOB'S DAYFILE, LISTING ALL THE CONTROL CARDS PROCESSED (EXCEPT CHARGE CARD) AND THE SYSTEM MESSAGES.

THE 6700, 6600 AND 6400 SYSTEMS OPERATE INDEPENDENTLY BUT THE OPERATING SYSTEMS ARE COMPATIBLE. TAPES AND DEVICE SETS (PRIVATE DISK PACKS) ARE SHARED BY THE 6600 AND 6700. IF A USER MUST HAVE THE SAME INFORMATION AT ALL COMPUTERS, HE SHOULD CREATE DUPLICATES, SO THAT JOBS ARE NOT ABORTED DUE TO THE UNAVAILABILITY OF SUCH FILES. THERE IS ONE PERMANENT FILE BASE SHARED BY THE 6600 AND 6700; THE 6400 HAS ITS OWN PERMANENT FILE BASE.

TO MOVE A TAPE FROM ONE BUILDING TO THE OTHER (6400) OR TO REMOVE A TAPE TEMPORARILY FROM THE COMPUTER SYSTEMS, THE USER MUST CONTACT THE TAPE LIBRARIAN, CODE 1896, BLDG 17, ROOM 100, (202) 227-1227.

POLICY GOVERNING THE USE OF THE COMPUTERS AND THE CHARGES FOR DIFFERENT CATEGORIES OF SERVICE IS GIVEN IN THE COMPUTER CENTER POLICY, OCTOBER 1977. INFORMATION ON CURRENT CHARGES MAY BE OBTAINED FROM CODE 1891, (202) 227-1361.

CLASSIFIED WORK IS RUN AT SPECIFIED TIMES WHEN NO REMOTE TERMINALS HAVE ACCESS TO THE CENTRAL COMPUTER.

*** PRIVACY ACT OF 1974 ***

PUBLIC LAW 93-579, WHICH BECAME EFFECTIVE IN THE FALL OF 1975, PROVIDES SAFEGUARDS FOR INDIVIDUALS AGAINST INVASION OF THEIR PERSONAL PRIVACY. THIS LAW ALSO PROVIDES SERIOUS INDIVIDUAL PENALTIES FOR VIOLATIONS. THE FOLLOWING SIMPLE SECURITY RULES AND PROCEDURES ARE THE RESPONSIBILITY OF EACH USER TO FOLLOW TO HELP MAINTAIN THE INTEGRITY OF THE COMPUTER SYSTEM'S FILES.

- 1) USERS MUST HAVE EXPLICIT PERMISSION BEFORE THEY ACCESS FILES, COMPUTER LISTINGS, CARDS, TAPES, DISK PACKS, DOCUMENTATION, ETC., WHICH BELONG TO OTHER INDIVIDUALS AND PROJECTS. (PUBLIC ACCESS FILES AS DOCUMENTED IN THIS MANUAL AND IN THE COMPUTER CENTER NOTES (SEE CCRM, 13-1) ARE AVAILABLE TO ALL USERS.)
- 2) JOB ORDER NUMBERS, USER ID'S, TAPE ID'S AND INTERCOM ID'S MUST BE USED ONLY WHEN AUTHORIZED.
- 3) USERS HAVE NO AUTHORITY TO TAMPER WITH THE OPERATING SYSTEM. PROGRAMS, ROUTINES AND FILES (PASSWORD, ACCOUNT, ETC.), USED FOR SYSTEM MAINTENANCE MAY NOT BE ACCESSED.
- 4) PERMANENT FILES AND TAPES, CONTAINING PERSONAL DATA, SHOULD BE PROTECTED WITH PASSWORDS (SEE CCRM, 3-6) AND LABELS, RESPECTIVELY, AND THESE IDENTIFICATIONS SHOULD BE KEPT CONFIDENTIAL AND CHANGED FREQUENTLY TO DISCOURAGE DISCLOSURE OF FILE DATA.
- 5) INTERCOM USERS SHOULD USE THE TURNKEY PASSWORD (CCRM, 9-2) TO PROTECT THEIR ACCOUNT NUMBERS.
- 6) FOR ASSISTANCE IN PASSWORD PROTECTION OF YOUR FILES OR CLARIFICATION OF ANY OF THE ABOVE, CONTACT USER SERVICES (202) 227-1907.
- 7) USERS WHO REQUIRE SPECIAL HANDLING FOR DATA TAPES AND OUTPUT CONTAINING PERSONAL DATA SHOULD NOTIFY OPERATIONS BRANCH, CODE 1806, (202) 227-1227.

SEE CCRM, 2-5, FOR A MEANS OF IDENTIFYING PRINTOUTS AS PRIVATE.

*** KEYPUNCH CHARACTERS ***

THE STANDARD KEYPUNCH CHARACTER SET IS THAT OF THE IBM 026 MODEL. HOWEVER, THERE ARE WAYS OF USING A SUBSET OF THE EBCDIC CHARACTER SET (SEE THE DTNSRDC CHARACTER SET, CCRM, 1-9 FOR THE CHARACTERS ALLOWED).

SOURCE CARDS PUNCHED FROM THE PUNCH FILE OF CDC 6000 WILL NORMALLY BE 026 WHETHER INPUT WAS 026 OR 029. THERE IS A ROUTINE FOR CONVERSION OF AN 026 DECK TO 029 (SEE CCLIB/P: CV29)

PROGRAMS RECEIVED ON 7-TRACK TAPE IN EBCDIC ARE NOT HANDLED BY THE AUTOMATIC CONVERSION, BUT MUST BE CONVERTED BY A SPECIAL PROGRAM.

*** USE OF ALTERNATE KEYPUNCH CHARACTERS ***

FOR JOBS SUBMITTED AT CENTRAL SITE OR FROM A 1700

THE 029 KEYPUNCH MAY BE USED TO PUNCH CARDS FOR THE 6000, IF THE END-OF-RECORD CARD (OR THE JOB CARD) PRECEDING THOSE CARDS HAS 29 PUNCHED IN COLUMNS 79 AND 80. FOR EXAMPLE, IF ALL CARDS IN A JOB HAVE BEEN PUNCHED ON AN 029 KEYPUNCH, THEN 29 SHOULD BE PUNCHED IN THE JOB CARD. HOWEVER, IF ONLY THE SOURCE CARDS HAVE BEEN PUNCHED WITH THE EBCDIC CHARACTERS, THEN A 29 SHOULD BE PUNCHED IN COLUMNS 79 AND 80 OF THE END-OF-RECORD CARD (7/8/9) IMMEDIATELY PRECEDING THE SOURCE CARDS. THE INPUT FOR A SUCCEEDING RECORD MAY BE RESET TO 026 MODE BY PUNCHING 26 IN COLUMNS 79 AND 80 OF THE PRECEDING 7/8/9 END-OF-RECORD. NOTE THAT ALL CARDS IN A LOGICAL RECORD MUST BE OF THE SAME KEYPUNCH CHARACTER SET. (SEE CCRM, 1-9 AND APPENDIX A OF ANY REFERENCE MANUAL)

THE MOST FREQUENTLY USED CHARACTERS WHICH DIFFER BETWEEN 026 AND 029 KEYPUNCHES ARE THE FOLLOWING FOUR CHARACTERS () = AND +.

FOR JOBS SUBMITTED FROM A 200 USER TERMINAL

TO BE READ BY A 200 USER TERMINAL, ALL CARDS IN A JOB DECK MUST BE IN THE SAME PUNCHING, AND THE OPERATOR MANUALLY WILL SELECT 026 OR 029 MODE. THE 026 MODE IS STANDARD. (PUNCH 29 IN COLUMNS 79 AND 80 OF THE JOB CARD TO INDICATE THE MODE TO THE OPERATOR.)

*** INTERNAL DATA STRUCTURE ***

1. INTERNAL REPRESENTATION OF HOLLERITH DATA IS DISPLAY CODE IN THE CDC 6000 BUT INTERNAL BCD IN SC 4020 OR IBM 7090. THIS MEANS THAT PROGRAMS USING OCTAL CONSTANTS TO DEFINE HOLLERITH DATA WILL REQUIRE MODIFICATION. NOTE THAT A DISPLAY CODE ZERO IS AN OCTAL 33 AND NOT 00, WHICH COULD CAUSE TROUBLE WITH MASKS.

HOLLERITH WORD

INTERNAL MACHINE REPRESENTATION

 BLANKS
 ZEROES
 'TAPE48'

 55555555555555555555
 33333333333333333333
 24012005374355555555

2. THE CHARACTER SEQUENCE ON THE DTNSRDC CDC 6000 IS THE ASCII 63-CHARACTER SET (SEE CCRM, 1-9). NOTE THAT THE ALPHABET PRECEDES NUMBERS AND SPECIAL CHARACTERS FOR FORTRAN ALPHABETIC COMPARE (FTN, APPENDIX A). COBOL USES THE ASCII6 COLLATING SEQUENCE (NUMBERS BEFORE LETTERS) (COBOL, APPENDIX A). TO USE THE COBOL6 COLLATING SEQUENCE (LETTERS BEFORE NUMBERS), THE U PARAMETER MUST BE INCLUDED ON THE COBOL CARD (CCRM, 5-2). THE COLLATING SEQUENCE FOR SORT/MERGE IS SPECIFIED BY THE USER IN THE SORT/MERGE CONTROL CARDS (SORT, 4-10 THRU 14, APPENDIX B). SORTS USING SCOPE 3.3 DIRECTIVES ALWAYS USE ASCII6.
3. WORD LENGTH IN THE 6000 IS 60 BITS (10 BCD CHARACTERS PER WORD) AS COMPARED WITH 36 BITS (6 BCD CHARACTERS PER WORD) IN THE UNIVAC 1108 AND IBM 7090 AND 32 BITS (4 EBCDIC CHARACTERS PER WORD) IN THE IBM S/360. THIS AFFECTS THE CONVERSION OF PROGRAMS IN FOUR AREAS:
 - A. THE DEGREE OF PRECISION OF OPERATIONS IS MUCH HIGHER. THEREFORE CONVERGENCE FACTORS MAY BE SMALLER IN ABSOLUTE VALUE.
 - B. CONSTANTS AND DATA MAY BE INCREASED TO GREATER ACCURACY BY ADDING ADDITIONAL SIGNIFICANT DIGITS, UP TO 15 DIGITS.
 - C. OCTAL CONSTANTS USED IN MASKING OPERATIONS ARE GENERALLY AFFECTED AND REQUIRE ALTERATION ACCORDING TO THEIR INTENDED USE.
 - D. SINCE WORDS CONTAIN 10 CHARACTERS PER WORD INSTEAD OF 6 OR 4, DATA STATEMENTS THAT STORE A STRING OF HOLLERITH CHARACTERS MAY POSITION THE CHARACTERS IN DIFFERENT RELATIVE POSITIONS IN DIFFERENT WORDS, EXCEPT FOR THE FIRST 6 OR 4 CHARACTERS. ALL VARIABLE FORMATS (WHETHER READ IN FROM CARDS OR CREATED BY THE PROGRAMMER) SHOULD BE CHECKED.

4. THE 6000 USES SOME SPECIAL BIT CONFIGURATIONS IN FLOATING POINT ARITHMETIC TO INDICATE INDEFINITE AND INFINITE OPERANDS. AN INDEFINITE MAY BE CAUSED BY 0/0 AND AN INFINITE BY N/0 OPERATION. EITHER ERROR COULD BE CAUSED BY REFERENCING PROGRAM AREAS NOT INITIALIZED OR AREAS OVERWRITTEN DUE TO INADEQUATE STORAGE RESERVATION. THE CPU WILL NOT DO ANY FURTHER CALCULATION IF IT ENCOUNTERS SUCH A NUMBER AND THE JOB WILL ABORT WITH AN ERROR MODE 2 OR 4 (SEE ERROR MESSAGES, CCRM, 2-21).

```

+ INFINITY  3777XXXXXXXXXXXXXXXXXX
- INFINITY  4000XXXXXXXXXXXXXXXXXX
+ INDEFINITE 1777XXXXXXXXXXXXXXXXXX
- INDEFINITE 6000XXXXXXXXXXXXXXXXXX
            WHERE 'X' IS ANY OCTAL DIGIT, USUALLY 0.

```

5. THE WORD FORMAT OF FLOATING POINT NUMBERS HAS 48 BITS FOR THE COEFFICIENT, 11 BITS FOR THE EXPONENT PLUS 1 SIGN BIT. THE RANGE OF PERMISSIBLE NUMBERS IS $10^{**}-293$ TO $10^{**}322$. INTEGERS MAY BE 60 BITS BUT MUST NOT EXCEED 48 BITS ($2^{**}48 - 1$, 15 DIGITS) IF THEY ARE TO BE USED IN MULTIPLICATION AND DIVISION OPERATIONS. A HARDWARE MULTIPLY INSTRUCTION IS AVAILABLE IN NOS/BE 1.0. INTEGER DIVISION IS PERFORMED BY SOFTWARE USING THE FLOATING POINT REGISTERS.

| EXAMPLE | CDC 6000 | IBM 360 | IBM 7090 |
|---------|-----------------------|----------|--------------|
| 1 | 0000000000 0000000001 | 00000001 | 000000000001 |
| -1 | 7777777777 7777777776 | FFFFFFFF | 400000000001 |
| 1.0 | 1720400000 0000000000 | 41100000 | 201400000000 |
| -1.0 | 6057377777 7777777777 | C1100000 | 601400000000 |
| 2.0 | 1721400000 0000000000 | 41200000 | 202400000000 |
| 4.0 | 1722400000 0000000000 | 41400000 | 203400000000 |

NOTE THE DIFFERENCE IN FORMAT OF NEGATIVE NUMBERS:

| CDC 6000 | IBM 360 | IBM 7090 |
|---------------------------------------|---------------------------------------|------------------|
| ONE'S COMPLEMENT OF ABSOLUTE VALUE | TWO'S COMPLEMENT OF ABSOLUTE VALUE | SIGNED MAGNITUDE |

6. LOGICAL VARIABLES IN THE 6000 ARE EXPRESSED AS -1 FOR TRUE AND +0 FOR FALSE.
7. THE PROGRAM FIELD LENGTH IN CENTRAL MEMORY IS NOT ALWAYS SET TO ZERO BY THE LOADER. SEE PRESET/PRESETA OPTION ON LDSET CARD, CCRM, 2-16.
8. COMPUTER INSTRUCTIONS ARE 15 OR 30 BITS LONG. HENCE ONE CM WORD MAY CONTAIN UP TO FOUR INSTRUCTIONS. THE NO-OPERATION INSTRUCTION (46000) IS USED TO FILL WORDS WHEN THE NEXT INSTRUCTION CANNOT FIT INTO THE SAME WORD.

*** DTNSRDC CHARACTER SET ***

| DISPLAY CODE | CHARACTER | PUNCH 026 | PUNCH 029 IF DIFF | 7-TRACK TAPE | NOTE NAME |
|-----------------|-----------|--------------|----------------------------|-----------------|-------------------|
| 01 | AAA | 12-1 | | 61 | |
| 02 | BBB | 12-2 | | 62 | |
| 03 | CCC | 12-3 | | 63 | |
| 04 | DDD | 12-4 | | 64 | |
| 05 | EEE | 12-5 | | 65 | |
| 06 | FFF | 12-6 | | 66 | |
| 07 | GGG | 12-7 | | 67 | |
| 10 | HHH | 12-8 | | 70 | |
| 11 | III | 12-9 | | 71 | |
| 12 | JJJ | 11-1 | | 41 | |
| 13 | KKK | 11-2 | | 42 | |
| 14 | LLL | 11-3 | | 43 | |
| 15 | MMM | 11-4 | | 44 | |
| 16 | NNN | 11-5 | | 45 | |
| 17 | OOO | 11-6 | | 46 | |
| 20 | PPP | 11-7 | | 47 | |
| 21 | QQQ | 11-8 | | 50 | |
| 22 | RRR | 11-9 | | 51 | |
| 23 | SSS | 0-2 | | 22 | |
| 24 | TTT | 0-3 | | 23 | |
| 25 | UUU | 0-4 | | 24 | |
| 26 | VVV | 0-5 | | 25 | |
| 27 | WWW | 0-6 | | 26 | |
| 30 | XXX | 0-7 | | 27 | |
| 31 | YYY | 0-8 | | 30 | |
| 32 | ZZZ | 0-9 | | 31 | |
| 33 | 000 | 0 | | 12 | (SOMETIMES 00) |
| 34 | 111 | 1 | | 01 | |
| 35 | 222 | 2 | | 02 | |
| 36 | 333 | 3 | | 03 | |
| 37 | 444 | 4 | | 04 | |
| 40 | 555 | 5 | | 05 | |
| 41 | 666 | 6 | | 06 | |
| 42 | 777 | 7 | | 07 | |
| 43 | 888 | 8 | | 10 | |
| 44 | 999 | 9 | | 11 | |
| 45 | +++ | 12 | 12-6-8 | 60 | PLUS |
| 46 | --- | 11 | | 40 | MINUS |
| 47 | *** | 11-4-8 | | 54 | ASTERISK |
| 50 | /// | 0-1 | | 21 | SLASH |
| 51 | (((| 0-4-8 | 12-5-8 | 34 | LEFT PARENTHESIS |
| 52 |))) | 12-4-8 | 11-5-8 | 74 | RIGHT PARENTHESIS |
| 53 | \$\$\$ | 11-3-8 | | 53 | DOLLAR |
| 54 | === | 3-8 | 6-8 | 13 | EQUAL |
| 55 | | | | 20 | BLANK |
| 56 | ,,, | 0-3-8 | | 33 | COMMA |
| 57 | ... | 12-3-8 | | 73 | PERIOD |

| DISPLAY CODE | CHARACTER | PUNCH 026 | PUNCH 029 | 7-TRACK TAPE | NOTE NAME |
|-----------------|-----------|--------------|--------------|-----------------|-----------------------------|
| 60 | ### | 0-6-8 | 3-8 | 36 | POUND |
| 61 | {{{ | 7-8 | 12-2-8 | 17 | (1) L BRACKET |
| 62 | }}} | 0-2-8 | 11-2-8 | 32 | (1) R BRACKET |
| 63 | !!! | 2-8 | | | (2) COLON (SOMETIMES 16) |
| 64 | **** | 4-8 | 7-8 | 14 | QUOTE |
| 65 | --- | 0-5-8 | | 35 | UNDERLINE |
| 66 | !!! | 11-2-8 | 12-7-8 | 52 | (1) EXCLAMATION (3) |
| 66 | !!! | 11-0 | | 52 | (1) EXCLAMATION (3) |
| 67 | &&& | 0-7-8 | 12 | 37 | AMPERSAND |
| 70 | ''' | 11-5-8 | 5-8 | 55 | (1) APOSTROPHE |
| 71 | ??? | 11-6-8 | 0-7-8 | 56 | QUESTION |
| 72 | <<< | 12-2-8 | 12-4-8 | 72 | (1) LESS THAN (3) |
| 72 | <<< | 12-0 | | 72 | (1) LESS THAN (3) |
| 73 | >>> | 11-7-8 | 0-6-8 | 57 | GREATER |
| 74 | @@@ | 5-8 | 4-8 | 15 | (1) AT |
| 75 | \\ | 12-5-8 | 0-2-8 | 75 | REVERSE SLANT |
| 76 | ^^^ | 12-6-8 | 11-7-8 | 76 | CIRCUMFLEX |
| 77 | ;;; | 12-7-8 | 11-6-8 | 77 | (4) SEMICOLON |
| 55 | | 8-6 | 0-8-4 | | BLANK |

NOTES:

- (1) THE SIX CHARACTERS DIFFERENT BETWEEN SCOPE 3.3 AND 3.4 ARE LEFT BRACKET, RIGHT BRACKET, EXCLAMATION, APOSTROPHE, LESS THAN, AND AT. THE 026 PUNCH IS DIFFERENT FOR EACH. THE 029 PUNCH IS THE SAME EXCEPT FOR ALTERNATE PUNCHES 11-0 AND 12-0 (THE DISPLAY CODE ASSOCIATED WITH THE 029 IS DIFFERENT.) THESE 6 CHARACTERS MAY HAVE DIFFERENT GRAPHICS ON A 200UT. THERE IS NO CHANGE FROM SCOPE 3.4 TO NOS/BE 1.0.
- (2) ON 7-TRACK TAPE BECOMES ZERO (DISPLAY 33)
- (3) ALTERNATE PUNCHES
- (4) AVOID WHOLE WORD OF SEMICOLON (END-OF-RECORD)

*** OCTAL-DECIMAL CONVERSION ***

ADDRESSES AND DATA OCCURRING IN CORE DUMPS ARE OCTAL.

| OCTAL | DECIMAL |
|--------|---------|
| 10 | 8 |
| 40 | 32 |
| 100 | 64 |
| 400 | 256 |
| 1000 | 512 |
| 2000 | 1024 |
| 4000 | 2048 |
| 10000 | 4096 |
| 20000 | 8192 |
| 30000 | 12288 |
| 40000 | 16384 |
| 50000 | 20480 |
| 60000 | 24576 |
| 70000 | 28672 |
| 100000 | 32768 |
| 140000 | 49152 |
| 200000 | 65536 |

TABLES OF OCTAL-DECIMAL AND OCTAL-HEX ARE IN SUB, APPENDIX B.

*** CDC 6000 REAL FLOATING ***

| DECIMAL | POSITIVE | | NEGATIVE | |
|---------|------------|------------|------------|------------|
| .0001 | 1702643334 | 2726161031 | 6075134443 | 5051616746 |
| .0005 | 1705406111 | 5645706520 | 6072371666 | 2132071257 |
| .001 | 1706406111 | 5645706520 | 6071371666 | 2132071257 |
| .005 | 1710507534 | 1217270244 | 6067270243 | 6560507533 |
| .01 | 1711507534 | 1217270244 | 6066270243 | 6560507533 |
| .05 | 1713631463 | 1463146315 | 6064146314 | 6314631462 |
| .1 | 1714631463 | 1463146315 | 6063146314 | 6314631462 |
| .5 | 1717400000 | 0000000000 | 6060377777 | 7777777777 |
| 0.0 | 0000000000 | 0000000000 | 7777777777 | 7777777777 |
| 1. | 1720400000 | 0000000000 | 6057377777 | 7777777777 |
| 5. | 1722500000 | 0000000000 | 6055277777 | 7777777777 |
| 10. | 1723500000 | 0000000000 | 6054277777 | 7777777777 |
| 50. | 1725620000 | 0000000000 | 6052157777 | 7777777777 |
| 100. | 1726620000 | 0000000000 | 6051157777 | 7777777777 |
| 500. | 1730764000 | 0000000000 | 6047013777 | 7777777777 |
| 1000. | 1731764000 | 0000000000 | 6046013777 | 7777777777 |
| 5000. | 1734470400 | 0000000000 | 6043307377 | 7777777777 |
| 10000. | 1735470400 | 0000000000 | 6042307377 | 7777777777 |
| 1.0E15 | 2002706576 | 5114320000 | 5775071201 | 2663457777 |

*** OCTAL-DECIMAL FRACTIONS ***

| OCTAL | DEC | OCTAL | DEC |
|-------|------|-------|---------|
| .1 | .125 | .01 | .015625 |
| .2 | .250 | .02 | .031250 |
| .3 | .375 | .03 | .046875 |
| .4 | .500 | .04 | .062500 |
| .5 | .625 | .05 | .078125 |
| .6 | .750 | .06 | .093750 |
| .7 | .875 | .07 | .109375 |

| OCTAL | DEC | OCTAL | DEC |
|-------|---------|-------|---------|
| .001 | .001953 | .0001 | .000244 |
| .002 | .003906 | .0002 | .000488 |
| .003 | .005859 | .0003 | .000732 |
| .004 | .007812 | .0004 | .000976 |
| .005 | .009765 | .0005 | .001220 |
| .006 | .011718 | .0006 | .001464 |
| .007 | .013671 | .0007 | .001708 |

| OCTAL | DEC | OCTAL | DEC |
|--------|---------|---------|---------|
| .00001 | .000030 | .000001 | .000003 |
| .00002 | .000061 | .000002 | .000007 |
| .00003 | .000091 | .000003 | .000011 |
| .00004 | .000122 | .000004 | .000015 |
| .00005 | .000152 | .000005 | .000019 |
| .00006 | .000183 | .000006 | .000022 |
| .00007 | .000213 | .000007 | .000026 |

CONVERSION EXAMPLES

OCTAL NUMBER 1716 4135240123456773
 DECIMAL VALUE $2^{**}(-1) * .500$

.015625
 .005859
 .001220
 .000061
 .000015

 $2^{**}(-1) * .522780 = .26139$

OCTAL NUMBER 2002 7065765114320000
 DECIMAL VALUE $2^{**}62 * .875$

.000
 .011718
 .001220
 .000213
 .000022

 $2^{**}62 * .888173 = 1.0E15$

*** CONTROL CARD FIELD LENGTHS ***

DTNSRDC NOS/BE 1.0 USERS SHOULD OMIT CM FROM JOB CARD UNLESS CORE IN EXCESS OF 46000 WILL BE REQUIRED. THE SYSTEM ADJUSTS ITS FL AUTOMATICALLY FOR THE FOLLOWING PRODUCTS AND SYSTEM CONTROL CARDS:

| | |
|-----------------|-------|
| ATTACH | 2000 |
| AUDIT | 35000 |
| BBT06RM | 10000 |
| BEGIN, REVERT | 4500 |
| BKSP | 1000 |
| CATALOG | 2000 |
| CHARGE | 3200 |
| CLEAR | 200 |
| CKP | 13000 |
| COMBINE | 20000 |
| COMPARE | 7000 |
| COPY | 20000 |
| COPYBF, COPYBR | 20000 |
| COPYCF, COPYCR | 20000 |
| COPYCL | 37000 |
| COPYE | 6000 |
| COPYF | 6000 |
| COPYL, COPYLM | 12000 |
| COPYN | 6000 |
| COPYR | 6000 |
| COPYRM | 21000 |
| COPYSBF, COPYSF | 11000 |
| COPYSR | 11000 |
| CPINDEX | 15000 |
| DISPOSE | 200 |
| DUMPF | 26000 |
| ESTMATE | 37000 |
| EXTEND | 2000 |
| FILE | 2000 |
| ITEMIZE | 12000 |
| LABEL | 2000 |
| LIBRARY | 300 |
| LOADPF | 26000 |
| MOUNT | 2000 |
| PFDUMP | 35000 |
| PRNTSPY | 35000 |
| PURGE | 2000 |
| RANCONV | 6000 |
| RENAME | 2000 |
| REQUEST | 2000 |
| RESTART | 25000 |
| RETURN | 200 |
| REWIND | 200 |
| ROUTE | 1000 |
| SISTAT | 21000 |
| SKIPF, SKIPB | 1000 |
| SYSBULL | 2000 |
| TRANSF | 300 |
| UNLOAD | 200 |

*** RECOMMENDED FIELD LENGTHS ***

| | | | |
|-----------------------|--------|----------|---------------------------------------|
| ALGOL | 46000 | MINIMUM | SEE CCRM, 11-1 |
| APT | 65200 | | ATTACH,APT3. |
| | 122000 | | ATTACH,APT4. |
| BASIC | 40000 | | |
| CORCL | 61000 | MINIMUM | |
| COBOL5 | 65000 | MINIMUM | SEE CCRM, 5-15 |
| COMPASS | 55000 | | |
| DDL | 44000 | | |
| ECAP | 70000 | | ATTACH,ECAMP. |
| EDITLIR | 45000 | MINIMUM | |
| FORM | 53000 | MINIMUM | (MOST OPTIONS REQUIRE AT LEAST 60000) |
| FTN | 46000 | MINIMUM | (PREFERS 54000) |
| FTN,D | 61000 | MINIMUM | |
| FTN,CPT=2 | 54000 | MINIMUM | |
| GPSS | 70000 | MINIMUM | ATTACH,GPSS. |
| LOAD | 30000 | MINIMUM | (WILL GET MORE AS NEEDED) |
| MARS VI | 60000 | (UPDATES | REQUIRE 65000) SEE CCRM, 11-8 |
| MIMIC | 70000 | | ATTACH,MIMIC. |
| MNF | 53000 | | SEE CCRM, 4-26 |
| NETED | 13000 | | ATTACH,NETED. |
| OMNITAR | 170000 | | SEE CCRM, 11-17 |
| PERTC | 112000 | | ATTACH,PERTC. |
| PERT66 | 120000 | | ATTACH,PERT66,PERTTIME. |
| PL1 | 120000 | MINIMUM | ATTACH,PL1,ID=CSYS. |
| QU (QUERY UPDATE) | 65000 | MINIMUM | |
| RATFOR | 45000 | | ATTACH,RATFOR. |
| RUN | 45000 | | |
| SIMI5 (SIMSCRIPT I.5) | 57000 | | SEE CCRM, 11-19 |
| SIMI5 (" II.5) | | | SEE CCRM, 11-19 |
| SNOPCL | 76000 | MINIMUM | ATTACH,SNOBOL. |
| SORT/MERGE | 60000 | | |
| S2000 | 50000 | MINIMUM | SEE CCRM, 11-19 |
| UPDATE | 40000 | | |

NOTE: SOME PRODUCTS MAY REQUIRE EITHER CM OR BOTH CM AND REL.

FOR CATALOGED PROCEDURES SEE CCLIB AND CCLIB/P.

FOR ROUTINES ON UTILITY LIBRARY SEE CCRM, 10-20, AND CCLIB/U.

FOR SPECIAL PRODUCTS SEE CCRM, CHAPTER 11.

***** JOB PROCESSING CONTROL CARDS *****

*** INTRODUCTION ***

THE FOLLOWING JOB CARD REQUIREMENTS APPLY AT DTNSRDC SITE:

6700 - BLUE
6600 - LAVENDER
6400 - ORANGE

NO OTHER CARDS IN THE DECK SHOULD BE ANY OF THE ABOVE COLORS.
AT DTNSRDC SITE USER NAME AND CODE MUST BE ON JOB CARD AS COMMENTS.

SOME REMOTE SITES HAVE OTHER STANDARDS FOR COLOR OF CARDS USED OR
FOR NAME AND CODE INFORMATION PLACEMENT ON THE JOB CARD.

THE SECOND CARD MUST BE THE CHARGE CARD. OTHER CONTROL CARDS
FOLLOW.

ALL CONTROL CARDS BEGIN IN COLUMN 1 OF THE CARD AND UNLESS
OTHERWISE INDICATED HAVE NO EMBEDDED BLANKS. A PERIOD OR CLOSE
PARENTHESIS MUST TERMINATE THE CONTROL CARD PARAMETERS. EACH CONTROL
CARD IS A REQUEST FOR THE SYSTEM TO LOAD AND EXECUTE A PROGRAM OF THAT
NAME.

ON THE CDC NOS/BE SYSTEM ALL CONTROL CARDS ARE PLACED IN THE FIRST
RECORD (THAT IS, THE SEQUENCE OF CONTROL CARDS IS TERMINATED BY AN
END-OF-RECORD CARD, WHICH CONTAINS A 7/8/9 PUNCH IN COLUMN 1).

SUBSEQUENT RECORDS ARE DATA WHICH THE CONTROL CARDS MUST PROCESS.

A GREEN END-OF-FILE CARD (6/7/8/9 PUNCH IN COLUMN 1) WILL BE ADDED
AT END-OF-DECK BY ADP CONTROL IF NOT ALREADY PRESENT.

NO CHARACTERS ON PAGE 1-10 (DISPLAY CODE GREATER THAN 57) SHOULD BE
USED ON ANY CONTROL CARD OR MESSAGE TO THE OPERATOR.

REFUNDS ON LOST TIME ARE DEFINED IN POLICY. REQUESTS MUST BE
ACCOMPANIED BY OUTPUT OF THE RUN, USER TROUBLE FORM (CCRM, 13-1), AND
SHOULD BE FILED WITHIN FIVE WORKING DAYS. DECISIONS ON REFUNDS WILL
BE MADE BY CODE 189.

*** CONTROL CARD FORMATS ***

A CONTROL CARD HAS ONE OF THE FOLLOWING FORMS (HERE AND THROUGHOUT THIS MANUAL, WHEN A WORD OR PHRASE IS ENCLOSED IN <...> (E.G., <LIST>), THE <> ARE NOT PART OF THE STRING):

EXAMPLES

| | |
|-------------------------------|---------------------------------|
| PROG. | REDUCE, SUMMARY |
| PROG,<POSITIONAL-PARAMETERS>. | CHARGE, SKIPF |
| PROG,<KEYWORD-PARAMETERS>. | AUDIT, LDSET, UPDATE |
| PROG,<POSITIONAL,KEYWORD>. | CATALOG, DISPOSE, FILE |
| PROG,<LIST>. | RETURN, REWIND, SYSBULL, UNLOAD |

| | |
|-------------------------|---|
| 'PROG' | IS THE NAME OF THE PROGRAM OR COMMAND. |
| <POSITIONAL-PARAMETERS> | IS A STRING WHOSE ENTRIES MUST BE IN A SPECIFIC ORDER. OMITTED PARAMETERS ARE INDICATED BY 2 SUCCESSIVE COMMAS. |
| <KEYWORD-PARAMETERS> | IS A STRING WHOSE ENTRIES ARE IDENTIFIED BY NAME OR NAME=VALUE (E.G., AI=I, ID=XXXX) AND MAY BE IN ANY ORDER. |
| <POSITIONAL,KEYWORD> | IS A STRING OF BOTH POSITIONAL AND KEYWORD PARAMETERS. WHEN A CONTROL CARD HAS ANY POSITIONAL PARAMETERS, THEY MUST APPEAR BEFORE ANY KEYWORD PARAMETERS. |
| <LIST> | IS A STRING OF UNRELATED PARAMETERS WHOSE POSITION IS UNIMPORTANT. |

MOST CONTROL CARDS MAY NOT BE CONTINUED. THOSE WHICH MAY (E.G., ATTACH, CATALOG) ARE CONTINUED BY ENDING THE FIRST CARD AFTER A COMPLETE PARAMETER (INCLUDING THE COMMA) AND BEGINNING THE NEXT PARAMETER ON THE NEXT CARD.

THE JOB CARD IS THE FIRST CARD OF THE JOB AND DEFINES THE JOB NAME AND MACHINE REQUIREMENTS AS WELL AS THE PUNCH MODE (WHEN THE JOB IS A CARD DECK). IT HAS THE FORM:

```

JOBNAME      A 4- TO 7-CHARACTER NAME OF THE FORM XXXXXXXY
              XXXX - USER'S REGISTERED INITIALS
                  (CONTACT CODE 1891 OR 1892 TO REGISTER AS A
                  COMPUTER USER)
              YYY - SELECTED BY THE USER

```

NOTE: SOME REMOTE INSTALLATIONS MAY ALTER THE FORMAT OF THE JOBNAME AND/OR COMMENTS FIELD (9999 NAME).

IN THE FIELDS BELOW, THE IDENTIFYING LETTERS MUST PREFIX THE NUMBERS. UNNEEDED FIELDS MAY BE OMITTED, BUT NO EMBEDDED BLANKS. PARAMETERS FOLLOWING BLANKS ARE IGNORED AND DEFAULT VALUES ARE USED.

```

CM  MAXIMUM CENTRAL MEMORY (OCTAL) REQUIRED BY JOB.  MAY NOT EXCEED
    300000.  (DEFAULT: 46000)
T   TOTAL JOB TIME IN DECIMAL SECONDS, AT 6400 CPU RATE.
    (DEFAULT: 180)
IO  TOTAL IO TIME IN DECIMAL SECONDS.  (DEFAULT: INFINITY)
MT  NUMBER OF 7-TRACK MAGNETIC TAPE DRIVES REQUIRED BY JOB.
    (MAX: 3 ON 6700; 2 ON 6600/6400)
NT  NUMBER OF 9-TRACK MAGNETIC TAPE DRIVES REQUIRED BY JOB.
    (MAX: 2 ON 6700; 0 ON 6600/6400)
P   PRIORITY OF JOB                                ELIGIBILITY
    4   EXPRESS                                CM<100000 WITH T<200.  (MAX: MT2)
    3   REGULAR                                CM<100000 WITH T<3600.  (DEFAULT)
    2   DEFERRED  OR  OVERNIGHT IF LESS THAN 2 HOURS WALL CLOCK
        EMERGENCY                                BY SPECIAL WRITTEN REQUEST ONLY, SEE POLICY
        (SEE POLICY FOR ADDITIONAL CM AND TIME COMBINATIONS FOR P3, P4)
CP  SELECT A SINGLE CPU FOR WHOLE 6700 JOB:  "A" FOR 6600, "B" FOR 6400
    THROUGHPUT IS USUALLY BETTER WHEN SYSTEM SELECTS OWN CPU.
    CPB IS MORE EFFICIENT FOR FORTRAN COMPILATIONS.
RP  NUMBER OF DEVICE SET DRIVES REQUIRED BY JOB
    (MAX: 1 IN PRIME TIME; 2 IN NON-PRIME TIME; NONE FOR P4)
D   USED WITH JOB DEPENDENCY  (SEE NOSRE, 4-86; SCOPE, 4-54)

```

KP COLUMNS 79-80
 CMITTED OR 26 - CONTROL CARDS (OR ENTIRE JOB) PUNCHED IN 026 MODE
 29 - CONTROL CARDS (OR ENTIRE JOB) PUNCHED IN 029 MODE

XXXX. IMPLIES XXXX000,CM46000,T180,IO77777,MT0,NT0,P3,RP0-

JOB CARD (CONTINUED)

JOBS WAITING TO BE RUN ARE DIVIDED INTO GROUPS ACCORDING TO REQUIRED HARDWARE (TAPES) AND PRIORITY (P). CM AND T PARAMETERS ON THE JOB CARD SPECIFY MAXIMUM CENTRAL MEMORY IN OCTAL AND CENTRAL PROCESSOR TIME IN DECIMAL 6400 SECONDS FOR EACH JOB. THESE PARAMETERS ALONG WITH P DETERMINE WHEN A JOB MAY BE SELECTED FOR EXECUTION. IF THE CM OR T PARAMETERS EXCEED THE ALLOWED SIZE FOR A GIVEN P GROUP THE P IS REDUCED. P4 JOBS IN THE INPUT QUEUE ARE INITIATED BEFORE ANY P3.

WITHIN PRIORITY GROUPS, JOBS WITH LOW CM AND LOW T ARE SELECTED FIRST SINCE EACH PRIORITY GROUP IS SUBDIVIDED. CORE SUBDIVISIONS OCCUR AFTER 46000, 60000, 100000 AND 120000. TIME SUBDIVISIONS OCCUR AT 40, 100 AND 200 SECONDS.

*** CHARGE CARD ***

FOR EACH JOB, AN ACCOUNTING ROUTINE VALIDATES THE USER AND COMPUTER ACCESS NUMBER TO VERIFY THAT THERE ARE FUNDS AVAILABLE AND THE USER IS AUTHORIZED TO UTILIZE THOSE FUNDS. TO OBTAIN A COMPUTER ACCESS NUMBER, A DTNSRDC JOB ORDER NUMBER MUST BE REGISTERED WITH OR OBTAINED FROM CODE 1891, (202) 227-1361.

THE CHARGE CARD IS THE SECOND CARD OF THE JOB AND HAS THE FORM:
CHARGE,XXXX,JJJJJJJJJJ,OP.

THE FIELDS FOLLOWING THE WORD CHARGE ARE POSITIONAL.

| | |
|------------|--|
| XXXX | USER'S DTNSRDC REGISTERED USER INITIALS (E.G., AMOS, CASG) |
| JJJJJJJJJJ | TEN CHARACTERS OF COMPUTER ACCESS NUMBER WITH NO BLANK, DASH OR MINUS |
| OP | TYPE OF RUN (OPTIONAL) |
| | CC CODE CHECK (DEFAULT) |
| | PR PRODUCTION |
| | RS RESEARCH |

*** PRIVATE ***

THE CONTROL CARD PRIVATE. MAY BE USED TO IDENTIFY PRIVATE MATERIAL (PRINTOUTS). EACH PRIVATE PRINTOUT WILL BE IDENTIFIED AS SUCH AT THE BEGINNING AND END OF THE PRINTOUT.

PRIVATE. CREATE A BANNER PAGE CONTAINING THE WORD PRIVATE. WHEN PRIVATE,LFN. LFN IS OMITTED, THE BANNER IS PUT ON FILE OUTPUT; A SIMILAR BANNER, SURROUNDED BY A BOX OF STARS, IS PUT INTO THE DAYFILE. THIS CARD SHOULD FOLLOW THE CHARGE CARD. WHEN LFN IS SPECIFIED, THE BANNER IS PUT ON FILE LFN; THERE IS NO DAYFILE MESSAGE. THE SECOND FORM SHOULD BE USED ONCE BEFORE ANYTHING IS WRITTEN ON LFN AND AGAIN AFTER LFN IS COMPLETE. LFN CAN THEN BE ROUTED OR DISPOSED TO A PRINTER.

AT CENTRAL SITE, PRINTOUTS WITH THESE BANNERS WILL NOT BE PLACED INTO THE "PUBLIC" OUTPUT BINS. THEY MUST BE OBTAINED AT THE DISPATCH WINDOW BY A PROPERLY IDENTIFIED AND AUTHORIZED REPRESENTATIVE. ANY SCRAP OUTPUT, RESULTING FROM FORMS ALIGNMENT, JAMS, ETC., WILL ALSO BE RETURNED WITH THE USABLE OUTPUT.

EXAMPLES

1. OUTPUT FILE IS TO BE PRIVATE

```
JOB CARD
CHARGE CARD
PRIVATE.
***
" 6/7/8/9 EOF
```

2. PRIVATE FILE PFILE IS TO BE DISPOSED

```
JOB CARD
CHARGE CARD
... ANY COMMANDS NOT USING PFILE MAY OCCUR HERE
PRIVATE,PFILE.
ATTACH,ANY,MYPRIVATEFILE,ID=XXXX,PW=READ.
COPYF,ANY,PFILE.
PRIVATE,PFILE.
ROUTE,PFILE,DC=PR,TID=C. OR DISPOSE,PFILE,PR=C.
***
" 6/7/8/9 EOF
```

*** OTHER NOS/BE CONTROL CARDS ***

- CLFAR. UNLOAD ALL FILES EXCEPT 'INPUT' AND 'OUTPUT'.
TO UNLOAD ALL FILES INCLUDING 'INPUT' AND/OR 'OUTPUT', IT
OR THEY MUST BE SPECIFICALLY LISTED. NOTE THAT 'INPUT'
WILL BE IGNORED IN BATCH.
- COMMENT. FOR INSERTION OF COMMENTS INTO CONTROL CARD RECORD.
IT WILL BE DISPLAYED IN DAYFILE AT OPERATOR'S CONSOLE.
DOES NOT FLASH FOR OPERATOR ACTION.
- DMP,LLLLLL.
DMP,FFFFFF,LLLLLL.
DUMP FROM RELATIVE OCTAL ADDRESS FFFFFFF THRU LLLLLL.
IF FFFFFFF IS OMITTED, 0 IS USED. WILL STOP AT CURRENT
FIELD LENGTH (FL).
- EXIT. WHEN THERE IS AN ABNORMAL END TO PROGRAM EXECUTION,
THE CONTROL CARDS FOLLOWING THE EXIT CARD WILL BE EXECUTED
FOR EXAMPLE, EXIT.
DMP(0,25000)
- EXIT,S. IF IT IS DESIRED TO EXECUTE ADDITIONAL CARDS AFTER AN
ABNORMAL ENDING DURING COMPILATION OR SIMPLE CONTROL CARDS
WITH ILLEGAL POSITIONAL PARAMETERS, THESE CARDS MUST
FOLLOW AN 'EXIT,S.'. COMPILER ERRORS WILL NOT CAUSE A
BRANCH TO 'EXIT.'. ANY ERROR WHICH BRANCHES TO 'EXIT.'
WILL EXECUTE 'EXIT,S.' IF IT IS ENCOUNTERED FIRST.
- EXIT,U. IF IT IS DESIRED TO EXECUTE ADDITIONAL CARDS REGARDLESS OF
WHETHER THE JOB RAN OR NOT, THESE CARDS MUST FOLLOW AN
'EXIT,U.'.
- LIMIT,SIZE. WHERE SIZE IS OCTAL NUMBER OF BLOCKS OF 64 PRU'S (4096
WORDS). DEFAULT MASS STORAGE IS 1600 OCTAL BLOCKS (57,344
PRU'S). THE MAXIMUM IS 13000 OCTAL BLOCKS OR 1,300,000
OCTAL (360,448) PRU'S (LIMIT,13000.). TO KEEP A JOB FROM
GENERATING TOO MUCH OUTPUT, THE LIMIT MAY BE SET LOWER.
THE TOTAL ALLOWED MASS STORAGE IN USE IN A JOB INCLUDES
INPUT FILES, OUTPUT FILES, ATTACHED PERMANENT FILES, AND
SCRATCH FILES. RETURN OR UNLOAD MAY BE USED TO DECREMENT
THE MASS STORAGE IN USE. DISPOSE CARDS DO NOT. INPUT
FILES ARE ABOUT EIGHT CARDS PER PRU, AND OUTPUT FILES ARE
FIVE TO TEN LINES PER PRU.
- MODE,N. MODE ERROR BYPASS IS NOT FOR USE AT DTNSRDC. ATTEMPT TO
IGNORE ERROR MODE 1 MAY CAUSE MODE 0.

PRNTSPY(LFN) PRINTS OUT THE RESULTS OF SPY. IT READS THE TEMPORARY FILE 'DOSSIER' AND THE FILE 'LFN' CONTAINING THE RELOCATABLE BINARY PROGRAM UNDER SCRUTINY. IF LFN IS SPECIFIED, THE LISTED ADDRESSES ARE RELATIVE TO EACH SUBPROGRAM; IF OMITTED, THEY ARE RELATIVE TO RA. IF MULTIPLE SPY RUNS ARE MADE IN ONE JOB, 'RETURN,DOSSIER.' MUST BE SPECIFIED AFTER EACH 'PRNTSPY' TO AVOID DUPLICATE OUTPUT. (SEE CCRM, 4-16; SPY; CONV, CHAPTER 11)

REQUEST,LFN,*Q.

PUT LFN ON A QUEUE DEVICE. USED BEFORE CREATING A FILE TO BE ROUTED (CCRM, 2-10) OR DISPOSED (CCRM, 2-8). FOR OTHER FORMS OF THE REQUEST CARD, SEE CCRM, 3-5, 3-11, 3-17.

REWIND(LFN1,LFN2,...,LFNN)

REWIND(LFN) REWIND DISK OR TAPE FILES

UNLOAD(LFN1,LFN2,...,LFNN)

UNLOAD(LFN) REWIND AND UNLOAD TAPE SO OPERATOR MAY REMOVE IT FROM THE TAPE DRIVE. THE JOB CARD MT/NT COUNT IS NOT DECREASED, SO ANOTHER TAPE MAY BE MOUNTED.

RETURN(LFN1,LFN2,...,LFNN)

RETURN(LFN) REWIND AND UNLOAD TAPE AND RETURN TAPE DRIVE TO SYSTEM FOR REASSIGNMENT. SHOULD BE USED AS SOON AS TAPE USAGE IS COMPLETED. THE JOB CARD MT/NT COUNT IS DECREASED.

TO DETACH A DISK FILE FROM USER JOB, EITHER RETURN OR UNLOAD.

DO NOT USE BOTH RETURN AND UNLOAD ON THE SAME TAPE.

AT END-OF-JOB, ALL DISK FILES ARE AUTOMATICALLY RETURNED; TAPES ARE UNLOADED AND RETURNED.

SUMMARY.

THE ACCOUNTING SUMMARY, UP TO THE CURRENT POINT IN THE JOB, IS PUT INTO THE USER'S DAYFILE.

SYSBULL(<BLIST>) PRINT THE SYSTEM BULLETIN(S) SPECIFIED.

THREE BULLETINS ALWAYS EXIST:

INDEX - LIST OF ALL BULLETINS CURRENTLY AVAILABLE
(DEFAULT IF <BLIST> OMITTED)

BATCH - PRINTED AFTER THE BANNER PAGE OF ALL BATCH JOBS
LOGIN - TYPED AT INTERCOM LOGIN TIME

'BATCH' AND 'LOGIN' MAY REFER TO OTHER BULLETINS WHICH THE USER SHOULD PRINT. 'SYSBULL,ALL.' WILL PRINT ALL BULLETINS.

TRANSF,XXXXX.

SUBTRACT 1 FROM THE D COUNTER FOR JOB XXXXX IN DEPENDENCY SET. IF COUNTER IS THEN 0 IT MAY EXECUTE.

*** DISPOSE ***

DISPOSE(LFN,X=KY) IN ORDER TO RELEASE FILES, SUCH AS OUTPUT, BEFORE A JOB IS TERMINATED, THE DISPOSE CARD MAY BE USED. IT MAY BE USED TO SPECIFY SPECIAL PRINT/PUNCH FORMS. SEE ALSO 'ROUTE' (CCRM, 2-10).

LFN IS NAME OF THE LOGICAL FILE TO BE RELEASED (IF PF NAME, THE CARD WILL BE IGNORED) WHEN X IS OMITTED, CARD ACTS AS A RETURN. LFN MUST BE ON A QUEUE DEVICE (SEE CCRM, 2-7).

X INDICATES TYPE OF OUTPUT DEVICE (SEE NOSBE, 4-27; SCOPE, 4-51) SUCH AS
 PR FOR PRINTER
 PU FOR PUNCHED HOLLERITH CARDS
 PB FOR PUNCHED BINARY CARDS
 P8 FOR PUNCHED 80-COLUMN BINARY
 PT FOR 1700 CALCOMP PLOTTER
 TA FOR PAPER TAPE ASCII (1700 ONLY)
 TP FOR PAPER TAPE BINARY (1700 ONLY)
 (NOTE: PAPER TAPE FILES MAY NOT BE DISPOSED TO CENTRAL SITE.)

K MAY BE ONE OF
 C - DISPOSE TO CENTRAL SITE
 I - DISPOSE TO INTERCOM TERMINAL (1700, 200, TTY)

Y WHEN K=C, Y MAY BE ONE OF THE CHARACTER PAIRS ON PAGE 2-9 TO SPECIFY SPECIAL FORMS. WHEN Y IS OMITTED, OUTPUT IS ON STANDARD FORMS.
 WHEN K=I, Y MUST BE A 2-CHARACTER TERMINAL ID (CCRM, 12-1). ONLY PR (PRINT) OUTPUT MAY BE DISPOSED TO A TTY.

IF KY OMITTED, THE FILE IS DISPOSED TO THE SAME USER (INTERCOM ONLY)

DISPOSE,LFN. WILL EVICT FILE LFN FROM THE SYSTEM. IF FILE WAS OUTPUT, PUNCH OR PUNCHB, IT IS LOST (NOT PRINTED OR PUNCHED).

EXAMPLES

| | |
|------------------------|--|
| DISPOSE,OUTPUT,PR=C. | CURRENT CONTENTS OF FILE OUTPUT GO IMMEDIATELY INTO CENTRAL SITE PRINT QUEUE. WILL PRINT ON ONE PART UNLINED PAPER, DO NOT USE C1P FOR THIS. |
| DISPOSE,ANYFIL,PR=C3P. | PRINT ON THREE-PART CARBON AT CENTRAL. |
| DISPOSE,OUTPUT,PR=IAF. | CURRENT CONTENTS OF FILE OUTPUT GO IMMEDIATELY INTO PRINT QUEUE FOR NAVSEC. EVEN IF THIS IS LAST CONTROL CARD, DAYFILE IS NOT DISPOSED. |
| DISPOSE,MYFILE,PR=IYB. | SEND PRINT FILE TO 200UT WITH ID YB. |
| DISPOSE,ANYFIL,PR=IEF. | CREATE REMOTE OUTPUT FILE AT TERMINAL EF. TELETYPE USER MUST USE BATCH,LFN,LOCAL TO RETRIEVE FILE FOR LISTING. |
| DISPOSE,ANYFIL,PU=C. | PUNCH 300 DECK FROM ANYFIL AT CENTRAL SITE. |
| DISPOSE,SOMFIL,PU=IAE. | PUNCH DECK FROM SOMFIL AT ANNAPOLIS 1700. |
| DISPOSE,MYFILE,PR. | INTERCOM ONLY. CREATE REMOTE OUTPUT FILE AT SAME TERMINAL. |

SPECIAL FORMS CODES

Y FOR SPECIAL PRINT FORMS SHOULD BE ONE OF THE CHARACTERS PAIRS:

| | |
|----|--|
| 1P | ONE-PART UNLINED PAPER |
| 2P | TWO-PART PLAIN |
| 3P | THREE-PART PLAIN |
| 4P | FOUR-PART PLAIN |
| 1L | ONE-PART LINED (14 7/8 BY 11) |
| 2L | TWO-PART LINED |
| 3L | THREE-PART LINED |
| 4L | FOUR-PART LINED |
| 1C | ONE-PART CONFIDENTIAL |
| 1S | ONE-PART SECRET |
| 1T | ONE-PART TELETYPE (NARROW) (≤72 COLUMNS) |
| 3T | THREE-PART TELETYPE (NARROW) (≤72 COLUMNS) |
| MU | BLANK MULTILITH (14 7/8 BY 10) |
| BS | REPORT DISTRIBUTION ENVELOPES |
| BL | LABELS (2 ACROSS) SUPPLY |
| CD | 3 X 5 CARDS |
| PA | FOUR-PART PAYROLL |
| X2 | NARROW MULTILITH (10 BY 12) 10 BY 12 LOOP |
| X3 | STD 1080 FORM (6-PART) |
| X4 | LABELS (3 ACROSS) |
| 3F | PAYROLL FORM |
| ED | PAYROLL FORM (EARNING AND DEDUCTION) |
| W2 | PAYROLL FORM (IRS W-2) |
| X1 | PAYROLL FORM (FICA) |

Y FOR SPECIAL PUNCH CARD STOCK SHOULD BE ONE OF:

| | |
|----|-------------------------------------|
| BS | TRANSACTION CARDS |
| 2H | ANNAPOLIS TRANSACTION CARDS (CLOCK) |
| NS | SUPPLY RECEIPT |
| N1 | SUPPLY ISSUE |
| N2 | SUPPLY REQUISITION |
| N4 | MILESTONE |

ADDITIONAL SPECIAL FORMS MAY BE ASSIGNED IF FREQUENTLY USED.

*** ROUTE ***

THE ROUTE CONTROL CARD IS USED FOR DIRECTING THE DISPOSITION AND DEFINING THE CHARACTERISTICS OF A FILE.

THE ROUTE DISPOSITION DIRECTIVES INCLUDE SELECTION OF THE TERMINAL TO PROCESS THE OUTPUT AND SPECIFYING THE FORMS ON WHICH THE FILE IS TO BE OUTPUT. (THESE ARE ALSO AVAILABLE, TO A DEGREE, ON THE 'DISPOSE' CONTROL CARD.) NOTE THAT WITH THE ROUTE COMMAND, SPECIAL FORMS (SUCH AS MULTIPLE-PART PAPER) MAY NOT BE SPECIFIED AT REMOTE SITES (SEE CCRM, 14-3 FOR 'PM'). IN ADDITION, THE CHARACTERISTICS PARAMETERS DEFINE THE CODING (INTERNAL) AND THE PRINT TRAIN OR PUNCH CODE TO BE USED.

THE ROUTE COMMAND HAS THE FORM:

ROUTE, LFN, <PARAMETERS>.

THE FILE BEING ROUTED MUST BE ON A QUEUE DEVICE. IF NOT, 'ROUTE' WILL COPY IT BEFORE ROUTING. FOR BETTER PERFORMANCE, USE A 'REQUEST, LFN, *Q' CARD (CCRM, 2-7) BEFORE CREATING THE FILE. ALL PARAMETERS ARE OPTIONAL. IN GENERAL, IF A PARAMETER IS OMITTED, IT RETAINS THE DEFINITION FROM THE LAST ROUTE CARD WHICH REFERENCED THAT LFN (THE EXCEPTION IS THE 'DEF' PARAMETER).

ROUTE PARAMETERS

DISPOSITION CODE

DC=XX FILE DESCRIPTION, WHERE XX IS ONE OF
 FL - FILMPL
 FR - FILMPR
 IN - PLACE THE FILE IN THE INPUT QUEUE
 PR - ANY PRINTER
 PT - 1700 ON-LINE PLOT
 PU - PUNCH
 SC - EVICT (SCRATCH) THE FILE

THE KEYWORDS FID, FC ARE IGNORED WITH NO WARNING MESSAGE WHEN DC=IN. IF DC=IN AND DEF ARE BOTH SPECIFIED, THE ROUTE COMMAND IS NOT EXECUTED.

DC SAME AS DC=SC

ABSENT SAME AS PREVIOUS REFERENCE TO THIS LFN ON A ROUTE CARD. IF NONE, SAME AS DC=SC, EXCEPT FOR THE FOLLOWING SPECIAL FILE NAMES:

| SPECIAL LFN | ASSUMED DC | ASSUMED EC |
|-------------|------------|---|
| FILMPL | FL | |
| FILMPR | FR | |
| OUTPUT | PR | |
| PLOT | PT | |
| PUNCH | PU | 026 (BCD - DEFAULT) |
| PUNCH | PU | 029 (WHEN EC=029 USED -- CENTRAL SITE ONLY) |
| PUNCHB | PU | SR (BINARY) |
| P80C | PU | 80COL (80-COLUMN BINARY) |

A DISPOSITION CODE NEED NOT BE SPECIFIED FOR A SPECIAL FILE NAME. DC=SC MAY BE USED TO PREVENT THE FILE FROM BEING PROCESSED.

DEFERRED FILE ROUTING (BATCH ONLY)

DEF FILE DISPOSITION IS DEFERRED UNTIL END OF JOB. DEF MUST BE SPECIFIED ON EACH ROUTE CARD IF END-OF-JOB PROCESSING IS INTENDED. DEF AND DC=IN MAY NOT BE USED TOGETHER.

ABSENT FILE DISPOSITION TAKES PLACE IMMEDIATELY. THE FILE IS NO LONGER AVAILABLE.

TERMINAL IDENTIFICATION

TID RETURN FILE TO JOB ORIGIN

TID=C OUTPUT THE FILE AT CENTRAL SITE

TID=AA ROUTE THE FILE TO INTERCOM TERMINAL WITH ID OF 'AA'

ABSENT SEE NOTE BELOW

FILE IDENTIFICATION

FID THE FILE NAME, WHILE IN THE OUTPUT QUEUE, IS THE SAME AS THE JOB NAME

FID=* AS ABOVE, ONLY THE 6TH AND 7TH POSITIONS ARE REPLACED BY 2 UNIQUE SEQUENCE NUMBERS. WHEN FID=* IS USED, THE FILE NAME SENT IS LISTED IN THE DAYFILE. WHEN ROUTING TO DC=IN, ONLY THE LAST 2 CHARACTERS OF THE LISTED FILE NAME HAVE MEANING.

ABSENT SEE NOTE BELOW

EXAMPLE: ASSUME FILENAME = JOBNAME = ABCDE3P.
ASSUME NEXT SEQUENCE NUMBER IS 4G.
AFTER FID -- ABCDE3P
AFTER FID=* -- ABCDE4G

FORMS CODE (CENTRAL SITE ONLY)

FC=WW USE SAME CODE FOR PRINTER AND PUNCHED OUTPUT AS GIVEN FOR 'DISPOSE' COMMAND ON PAGE 2-9

FC USE STANDARD CARD OR PRINT FORMS

ABSENT SEE NOTE BELOW

REPEAT COUNT (CENTRAL SITE AND 200-UT-TYPE ONLY)

REP=NN NN IS NUMBER OF EXTRA COPIES FOR OUTPUT FILES (DEFAULT: 0)

NOTE: WHEN FILE IDENTIFICATION, FORMS CODE AND/OR TERMINAL IDENTIFICATION IS OMITTED, IT IS THE SAME AS IN THE PREVIOUS REFERENCE TO THE LFN ON A ROUTE CARD. IF OMITTED ON THE FIRST/ ONLY ROUTE CARD FOR AN LFN, THE DEFAULTS ARE FID, FC, TID, RESPECTIVELY.

*** ROUTE EXAMPLES ***

1. SEND ENTIRE OUTPUT AND DAYFILE TO ANOTHER TERMINAL (EG, AF).

```

JOB CARD
CHARGE, XXXX, JJJJJJJJJJ.
ROUTE, OUTPUT, DEF, DC=PR, TID=AF.          SET TO PRINT AT END OF JOB
***
' 7/8/9      EOR
***
" 6/7/8/9    EOF

```

NOTE THAT THE FILE MAY BE REFERENCED BEFORE IT IS CREATED. THE ROUTING INFORMATION IS SAVED UNTIL THE FILE IS SENT TO THE (REMOTE) OUTPUT QUEUE.

2. PUT A JOB INTO THE INPUT QUEUE. (THIS IS SIMILAR TO THE INTERCOM 'BATCH' COMMAND.)

```

JOB CARD
CHARGE, XXXX, JJJJJJJJJJ.
COPYE, INPUT, FALCON.
ROUTE, FALCON, DC=IN.
' 7/8/9      EOR
JOB CARD
CHARGE, XXXX, JJJJJJJJJJ.
***
' 7/8/9      EOR
" 6/7/8/9    EOF

```

COMPLETE JOB
TO GO INTO
INPUT QUEUE

3. A USER WISHES TO PRINT FILE 'OUTPUT' ON 3-PART PLAIN PAPER.

```
ROUTE, OUTPUT, TID=C, FC=3P.          SEE PAGE 2-9
```

A USER MUST PUNCH FILE 'PUNCH' ON SPECIAL CARDS (EG, SUPPLY ISSUE).

```
ROUTE, PUNCH, TID=C, FC=N1.          SEE PAGE 2-9
```

A USER WISHES TO PUNCH FILE 'ANY' AS AN 029 DECK.

```
ROUTE, ANY, DC=PU, EC=029, TID=C.    (CENTRAL SITE)
BEGIN, CV29, , ANY, AA.              (REMOTE 1700 WITH TERMINAL ID 'AA')
```

4. SEND A COPY OF THE OUTPUT FILE TO ANOTHER TERMINAL (EG, AE).

```

JOB CARD
CHARGE, XXXX, JJJJJJJJJJ.
***
REWIND, OUTPUT.
COPYE, OUTPUT, ANY.
ROUTE, ANY, DC=PR, TID=AE.
' 7/8/9      EOR
***
" 6/7/8/9    EOF

```

***** LOADER *****

THE LOADER IS RESPONSIBLE FOR LOADING ALL PROGRAMS, RESOLVING ANY EXTERNAL REFERENCES, AND OPTIONALLY INITIATING EXECUTION.

ONCE LOADING OF A PROGRAM IS STARTED, NO OTHER CONTROL CARDS MAY INTERRUPT THE LOAD SEQUENCE. FOR INSTANCE, A 'LOAD,LFN.' CARD MAY ONLY BE FOLLOWED BY ANOTHER 'LOAD,LFN1.' OR ONE OF THE LOADER CONTROL CARDS (SEE CCRM, 2-15,16) OR 'MAP', 'REDUCE' (SEE CCRM, 2-17,18) OR OTHERS LISTED IN THE LOADER MANUAL.

A LINE AT THE END OF THE LOAD MAP OF A SIMPLE OR OVERLAY LOAD CONTAINS XXXXXXB CM USED . THIS IS THE MINIMUM FL NEEDED TO LOAD. A SIMILAR MESSAGE IN THE DAYFILE ALSO GIVES THE PROGRAM LENGTH. USE THIS INFORMATION TO SAVE CORE ON FUTURE RUNS. AT PRESENT, THIS INFORMATION IS NOT AVAILABLE WITH SEGLOAD.

*** TYPES OF LOADING ***

LOADING DIFFERS ACCORDING TO WHETHER THE INPUT IS ONE OR MORE OBJECT MODULES OR A SINGLE CORE IMAGE MODULE. LOADING OF OBJECT MODULES CAN INVOLVE OVERLAY OR SEGMENT GENERATION AND CAN RESULT IN ONE OR MORE CORE IMAGE MODULES. A BASIC LOAD RESULTS IN ONE CORE IMAGE (ABSOLUTE) MODULE.

| | |
|-----------------------|---|
| OBJECT MODULE LOADING | ONE OR MORE OBJECT MODULES ARE LOADED, LIBRARIES ARE SEARCHED FOR THE EXTERNAL REFERENCES, ADDRESSES ARE ADJUSTED, AND A CORE IMAGE MODULE MAY BE PRODUCED. |
| CORE IMAGE LOADING | THIS IS A SPECIAL CASE BECAUSE NO EXTERNAL LINKAGE OR ADDRESS ADJUSTMENT IS REQUIRED. |
| BASIC LOADING | ALL OBJECT CODE IS LOADED AT THE SAME TIME, RESULTING IN A SINGLE CORE IMAGE MODULE. |

SEGMENTATION

FOR LARGE PROGRAMS, SEGMENTATION SHOULD BE USED TO DIVIDE THE PROGRAM INTO SEVERAL CORE IMAGE MODULES, CALLED SEGMENTS.

WITH SEGMENTATION, ONLY THOSE PORTIONS OF THE PROGRAM NEEDED AT A GIVEN MOMENT ARE IN CORE. DIFFERENT CORE IMAGE MODULES RESIDE IN THE SAME AREA OF CORE AT DIFFERENT TIMES. DEPENDING ON EXECUTION REQUIREMENTS, DIFFERENT CORE IMAGE MODULES ARE LOADED DYNAMICALLY.

FEATURES:

- SEGMENTATION ALLOWS ANY NUMBER OF LEVELS, LIMITED ONLY TO A TOTAL OF 4093 SEGMENTS.
- AFTER SEGMENTS HAVE BEEN GENERATED, THEIR LOADING IS AUTOMATIC.
- REFERENCES BETWEEN SEGMENTS MAY BE UPWARD OR DOWNWARD.
- AT EXECUTION TIME, A RESIDENT PROGRAM IS LOADED WHICH LOADS THE ROOT SEGMENT. THEREAFTER, IT LOADS THE OTHER SEGMENTS AS REQUIRED.

OVERLAY GENERATION

PROVIDED PRIMARILY FOR COMPATABILITY WITH PREVIOUS LOADERS.

ADVANTAGES OF OVERLAY OVER SEGMENTATION:

- 1 - OVERLAY CORE IMAGE MODULES CAN BE HANDLED BY 'EDITLIB' WHEREAS SEGMENTS CANNOT.

DISADVANTAGES OF OVERLAY:

- 1 - ONLY 3 LEVELS OF OVERLAY ARE PERMITTED.
- 2 - USER MUST INSERT CALLS IN HIS PROGRAM TO LOAD SUBSEQUENT OVERLAYS.
- 3 - RESTRUCTURING OF OVERLAYS REQUIRES SOURCE CODE MODIFICATION. (SEGMENTATION STRUCTURE IS DEFINED OUTSIDE THE PROGRAM, THUS NO SOURCE CODE MODIFICATION IS NEEDED TO RESTRUCTURE SEGMENTS.)
- 4 - EACH OVERLAY MUST BEGIN WITH A MAIN PROGRAM STATEMENT. (SEGMENTATION HAS ONLY ONE MAIN PROGRAM.)
- 5 - ONLY SELF AND UPWARD REFERENCES ARE ALLOWED.
- 6 - ENTIRE LOADER MUST BE IN CORE, WHEREAS SEGMENTATION HAS ONLY A SMALL RESIDENT.

*** LOADER CONTROL CARDS ***

LOAD(LFN1,LFN2,...,LFNN)

FILES, WHOSE CONTENTS ARE TO BE LOADED, ARE SPECIFIED IN ONE OF THE FOLLOWING FORMS:

LFN - REWIND (UNLESS INPUT) BEFORE LOADING
LFN/R - REWIND BEFORE LOADING
LFN/NR - NO REWIND BEFORE LOADING

EXECUTE. COMPLETES LOADING, FILLS UNSATISFIED REFERENCES BY SYSTEM (AND USER) LIBRARY SEARCH, GENERATES LOAD MAP AND EXECUTES PROGRAM.

EXECUTE(PNAME,PLIST) TO BEGIN EXECUTION AT ENTRY POINT PNAME
SEE LOADER, 2-9; LOADER/S, 2-9.

NAME. LOAD AND EXECUTE BINARY PROGRAM IN LFN 'NAME'. FOR A
NAME(PLIST) MULTIPLE LOAD JOB, THE LAST FILE LOADED MAY USE THIS
FORMAT TO INITIATE EXECUTION AS WELL.
FOR EXAMPLE: LOAD(INPUT)

LGO.

PARAMETERS MAY BE SUPPLIED TO THE CALLED PROGRAM. FTN PL
LINE LIMIT MAY BE RESET (SEE CCRM, 4-2). FILE NAMES
DEFINED ON A FORTRAN PROGRAM CARD MAY BE OVERRIDDEN IN
SEQUENTIAL ORDER. FOR EXAMPLE: LGO(,NEWOUT)
SUBSTITUTES NEWOUT FOR THE THIRD FILE NAMED.
IF 'NAME' IS NOT A BINARY PROGRAM, AN OCTAL AND CHARACTER
DUMP OF THE FIRST 25 WORDS WILL APPEAR IN THE DAYFILE
ALONG WITH AN ERROR MESSAGE.

NOGC. COMPLETES LOADING OF A PROGRAM, INCLUDING GENERATING LOAD
NOGC(LFN) MAP, BUT BYPASSES EXECUTION. LFN WILL CONTAIN THE
LOADED PROGRAM AS A SINGLE CORE IMAGE MODULE
(NON-SEGMENTED/NON-OVERLAY LOADS ONLY). IF LFN IS
CATALOGED FOR PRODUCTION USE, THE ABSOLUTE PROGRAM WILL
REDUCE TO MINIMUM CORE IN EXECUTION. LFN IS SUITABLE FOR
INCLUSION IN A USER EDITLIB.

LIBLOAD(LIBNAME,EPTNAME1,EPTNAME2,...,EPTNAMEN)

WILL LOAD MODULES FROM SPECIFIED LIBRARY WHICH CONTAIN THE
SPECIFIED ENTRY POINTS. FOR CORE IMAGE LOAD, ONLY ONE
ENTRY POINT MAY BE GIVEN. SEE LOADER, 2-7; LOADER/S, 2-7.

SLOAD(LFN,NAME1,NAME2,...,NAMEN)

WILL SELECTIVELY LOAD MODULES FROM FILE LFN. FOR CORE
IMAGE LOAD, ONLY ONE ENTRY POINT MAY BE GIVEN.

LFN MAY HAVE THE FOLLOWING FORMS:

LFN - REWIND (EXCEPT INPUT) BEFORE LOAD
LFN/R - REWIND BEFORE LOAD
LFN/NR - NO REWIND BEFORE LOAD

SATISFY(LIBNAME1,LIBNAME2,...,LIBNAMEN)

SATISFY SATISFY UNSATISFIED EXTERNALS PRIOR TO NORMAL SATISFACTION
AT LOAD COMPLETION: SATISFY EXTERNALS FROM USER-SPECIFIED
LIBRARIES IN THE ORDER SPECIFIED. CANNOT BE USED IN
SEGMENTED LOADS.

*** LDSET ***

LDSET(OPTION1,OPTION2,...,OPTIONN)
WILL SET ANY OF SEVERAL OPTIONS FOR THE CURRENT LOAD ONLY.

EACH OPTION HAS ONE OF THE FOLLOWING FORMS:

KEY
KEY=PARAM
KEY=PARAM1/PARAM2/.../PARAMN

SOME OF THE OPTIONS ARE:

FILES=LFN1/LFN2/.../LFNN
PERMITS RECORD MANAGER USERS TO ASSURE THAT
SYSTEM ROUTINES ARE LOADED FOR THE
PROCESSING OF SPECIFIED FILES (CCRM, 3-18)

LIB=LIBNAME1/LIBNAME2/.../LIBNAMEN
SPECIFIES ONE OR MORE LIBRARIES COMPOSING
THE LOCAL LIBRARY SET. (CONTRAST WITH
'LIBRARY' CONTROL CARD.)

MAP=P/LFN SPECIFIED LOAD MAP GENERATION.
MAP=/LFN LFN IS THE FILE TO CONTAIN THE LOAD MAP.
MAP=P (DEFAULT IS 'OUTPUT')

P IS ONE OR MORE OF THE FOLLOWING LETTERS
(DEFAULT: SB):

N - NO MAP
S - ERROR MESSAGES AND LOADER STATISTICS
R - A BLOCK LIST AND A LIST OF UNSATISFIED
EXTERNALS
E - A LIST OF ENTRY POINTS (WITHOUT CROSS-
REFERENCE MAP)
X - A LIST OF ENTRY POINTS (WITH CROSS-
REFERENCE MAP)

PRESET=P PRESET CORE IN THE SPECIFIED MANNER.
PRESETA=P (PRESETA PUTS EACH LOCATION'S ADDRESS IN THE
LOW ORDER BITS.)

SOME VALUES OF P ARE:

| P | OCTAL PRESET VALUE |
|---------|--------------------------|
| NONE | NO PRESETTING |
| ZERO | 0000 0000 0000 0000 0000 |
| NGINF | 4000 0000 0000 0000 0000 |
| ALTZERO | 2525 2525 2525 2525 2525 |

(THIS IS ALTERNATING 0 AND 1 BITS)

P MAY ALSO BE A 1-20 DIGIT OCTAL CONSTANT
WITH OPTIONAL + OR - AND OPTIONAL B SUFFIX.

FOR ADDITIONAL VALUES, SEE LOADER, 2-17:
LOADER/S, 2-16.

USE=EPTNAM1/EPTNAM2
USEP=RTNAME1/RTNAME2

USED IN OVERLAY JOBS TO FORCE LOADING OF
CERTAIN RECORD MANAGER ROUTINES INTO OVERLAY
0,0 WHEN ERROR NE4340 OCCURS.

FOR MORE OPTIONS, SEE LOADER, 2-15 THRU 2-25; LOADER/S, 2-14 THRU 2-24.

*** LOADER-RELATED CONTROL CARDS ***

MAP(P) SPECIFY DEFAULT OPTION FOR LOAD MAPS. IT REMAINS IN EFFECT UNTIL CHANGED BY ANOTHER MAP STATEMENT. IT MAY BE OVERRIDDEN FOR THE NEXT LOAD BY USING THE MAP= OPTION OF THE 'LDSET' STATEMENT.

P HAS ONE OF THE FOLLOWING VALUES:

- OFF - NO MAP (SAME AS LDSET,MAP=0)
- PART - STARTING LOCATIONS OF PROGRAMS AND SUBPROGRAMS (SAME AS LDSET,MAP=SB.) (DEFAULT ON OUR SYSTEM)
- ON - FULL MAP WITH CROSS REFERENCES TO ALL ENTRY POINTS (SAME AS LDSET,MAP=SBX.). MORE CP TIME IS USED TO GENERATE A FULL RATHER THAN PARTIAL MAP.

'MAP' MAY OCCUR WITHIN A LOAD SEQUENCE.

LIBRARY(LIBNAME1,LIBNAME2,...,LIBNAMEN)
LIBRARY. SPECIFY A SET OF GLOBAL LIBRARIES TO BE SEARCHED FOR EXTERNALS AND PROGRAMS AND THE ORDER IN WHICH THE LIBRARIES ARE TO BE CONSIDERED. THIS SET REMAINS IN EFFECT UNTIL END-OF-JOB OR UNTIL ANOTHER 'LIBRARY' STATEMENT IS ENCOUNTERED.

THE ORDER OF SEARCH FOR EXTERNALS IS:

- GLOBAL LIBRARIES (THOSE ON LAST 'LIBRARY' STATEMENT)
- LOCAL LIBRARIES (THOSE IN LIB= PARAMETER OF 'LDSET')
- SYSTEM LIBRARY ('NUCLEUS')

THE ORDER OF SEARCH FOR PROGRAMS IS:

- LOCAL FILES
- GLOBAL LIBRARIES
- LOCAL LIBRARIES
- SYSTEM LIBRARY

THE MAXIMUM NUMBER OF USER LIBRARIES WHICH MAY BE SPECIFIED ON A 'LIBRARY' STATEMENT IS 2. THE FOLLOWING LIMITS ON THE NUMBER OF LIBRARIES SPECIFIED APPLY:

| USER | SYSTEM |
|------|--------|
| 0 | 24 |
| 1 | 13 |
| 2 | 2 |

'LIBRARY.' NULLIFIES THE EFFECT OF THE PREVIOUS 'LIBRARY' STATEMENT. 'LIBRARY' MAY NOT OCCUR WITHIN A LOAD SEQUENCE.

REDUCE. TURNS REDUCE FLAG ON. IT REMAINS ON UNTIL END OF JOB OR UNTIL AN 'RFL' CARD IS ENCOUNTERED. THE LOADER USES AS MUCH FL AS NEEDED FOR THE LOAD (UP TO JOB CARD CM), THEN JUST BEFORE PROGRAM EXECUTION, REDUCES FL TO MINIMUM NEEDED TO CONTAIN THE LOADED MODULES.

| TYPE LOAD | MINIMUM FL |
|-----------|---|
| ----- | ----- |
| BASIC | HIGHEST ADDRESS OF ALL PROGRAMS AND BLOCKS (INCLUDING BLANK COMMON) |
| OVERLAY | HIGHEST ADDRESS OF ALL OVERLAYS |
| SEGMENTED | FOR A PROGRAM WITH BLANK COMMON: BLANK COMMON FOLLOWS THE HIGHEST SEGMENT. MINIMUM FL IS THE END OF BLANK COMMON. FOR A PROGRAM WITHOUT BLANK COMMON BUT WITH MULTIPLE LEVELS: MINIMUM FL IS END OF HIGHEST SEGMENT IN HIGHEST LEVEL. FOR A PROGRAM WITHOUT BLANK COMMON AND ONLY ONE LEVEL: MINIMUM FL IS END OF ROOT SEGMENT. FL IS ADJUSTED DYNAMICALLY AS SEGMENTS ARE LOADED AND UNLOADED. |

IN ALL CASES, FL IS ROUNDED UP TO A MULTIPLE OF 100 OCTAL WORDS. 'REDUCE' MAY OCCUR WITHIN A LOAD SEQUENCE.

RFL,XXXXXX. REQUEST A NEW OCTAL FIELD LENGTH DURING THE JOB EXECUTION (UP TO JOB CARD CM). 'RFL' TURNS OFF THE REDUCE FLAG FOR THE NEXT LOADING SEQUENCE. THEREFORE, A 'REDUCE' CARD MUST BE USED AFTER THE 'RFL' TO TURN THE FLAG BACK ON, IF DESIRED.
'RFL' SETS THE MAXIMUM FL A PROGRAM MAY USE UNTIL ANOTHER 'RFL' IS ENCOUNTERED.

EXAMPLE:

| | |
|------------------------|--|
| REDUCE. | ** TURN ON REDUCE FLAG |
| LG01. | ** LOAD, REDUCE FL TO MINIMUM, EXECUTE |
| RFL,60000. | ** INCREASE FL FOR NEXT LOAD, TURN REDUCE FLAG OFF FOR NEXT LOAD |
| LG02. | ** LOAD AND RETAIN 60000B CM |
| ATTACH, LG03, ID=XXXX. | |
| LG03. | ** LOAD IN UP TO 60000 (LAST RFL), EXECUTE |
| RFL,70000. | ** INCREASE FL FOR NEXT LOAD, TURN REDUCE FLAG OFF FOR NEXT LOAD |
| REDUCE. | ** TURN REDUCE FLAG BACK ON |
| LG04. | ** LOAD IN UP TO 70000 (LAST RFL), REDUCE FL, EXECUTE |

'RFL' MAY NOT APPEAR WITHIN A LOAD SEQUENCE.

*** SEGMENTATION ***

TO IMPLEMENT SEGMENTATION A SEPARATE DIRECTIVE RECORD IS PREPARED TO DESCRIBE THE TREE STRUCTURE. THE MODULES WILL BE LOADED AUTOMATICALLY AS NEEDED. JOB FIELD LENGTH IS ADJUSTED DYNAMICALLY IF PROGRAM HAS NO BLANK COMMON, NO LEVEL CARDS AND NOT IN RFL MODE.

ALL NECESSARY RECORD MANAGER ROUTINES MUST BE IN ROOT SEGMENT.

SEGLOAD, I=LENDIR, B=LENABS.

INITIATE A SEGMENTATION JOB AND READ THE DIRECTIVE RECORD.

I=LENDIR - DIRECTIVES ARE ON FILE LENDIR
(DEFAULT: I=INPUT)

B=LENABS - BINARY ABSOLUTE MODULE WILL BE ON FILE
LENABS (DEFAULT: B=ABS)

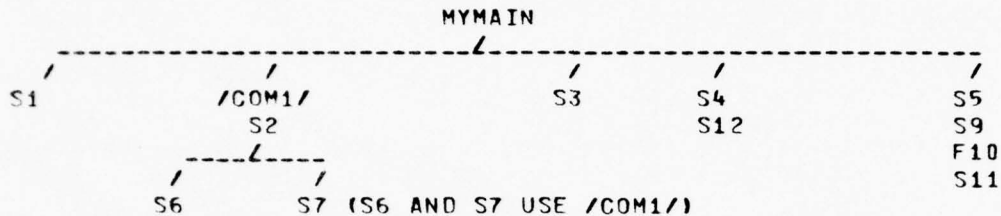
OTHER LOAD AND LDSET CARDS FOLLOW THIS CARD.

SEGLOAD DIRECTIVES

- X TREE Y TO DEFINE A TREE STRUCTURE.
Y MAY BE COMMA SEPARATED LIST OF OTHER TREES (PRE-DEFINED),
SEGMENTS OR NAMES OF INDIVIDUAL SUBPROGRAMS TO BE ASSIGNED
A COMMON STARTING ADDRESS.
- X TREE F-(C,D) TO INDICATE BRANCHING OF THE TREE USE -, THEN ALL
FOLLOWING ITEMS ARE ENCLOSED IN PARENTHESES.
- C INCLUDE A,B TO ASSIGN PROGRAMS A AND B TO SEGMENT C.
COPIES OF A ROUTINE MAY BE IN DIFFERENT SEGMENTS.
- C GLOBAL COM1,COM2 TO ESTABLISH NAMED COMMONS AT DESIRED SEGMENT.
REFERENCE NAME TO LEFT OF DIRECTIVE MUST BE DEFINED BY A
PREVIOUS DIRECTIVE IF SEGMENT.
- C GLOBAL COM1,COM2-SAVE TO SAVE GLOBAL BLOCK ON DISK FOR LATER CALLS
TO THE SEGMENT WHICH CONTAINS IT.
- END EPT SHOULD BE THE LAST DIRECTIVE IN THE RECORD, WHERE EPT IS
THE ENTRY OF THE MAIN PROGRAM IN THE ROOT SEGMENT.
NONFATAL ERROR IF OMITTED.

SAMPLE TREE DIAGRAM

A BLOCK DATA SUBPROGRAM DEFINES COMMON /COM1/ WHICH IS TO BE LOADED WITH PROGRAM S2. /COM1/ IS ALSO REFERRED TO BY S6 AND S7.



```

SEG INCLUDE S2,BLKDAT.
SEG GLOBAL COM1
PLUM TREE SEG-(S6,S7)
PEAR TREE MYMAIN-(S1,PLUM,S3,S4,S5)
S5 INCLUDE S9,F10,S11
S4 INCLUDE S12
END
  
```

BY USING NESTED PARENTHESES ONE TREE CARD MAY BE ELIMINATED.

*** SEGMENTATION CAUTIONS ***

TO DEVELOP A SEGMENTED JOB, SEVERAL RUNS MAY BE REQUIRED, SO RELOCATABLE OBJECT CODE SHOULD BE CATALOGED. COMMON BLOCKS AND RECORD MANAGER ROUTINES MAY NEED TO BE INCLUDED IN LOWER SEGMENTS TO OPERATE PROPERLY.

THE LOAD MAP MUST BE CHECKED CAREFULLY FOR ANY DUPLICATE COMMON BLOCK ENTRIES. EACH COMMON BLOCK WHICH IS REFERENCED IN MORE THAN ONE SEGMENT MUST BE PUT INTO A GLOBAL AT THE NEAREST-TO-THE-ROOT SEGMENT. IF ANY COMMON BLOCK APPEARS MORE THAN ONCE WITHOUT "SAFE", A GLOBAL IS REQUIRED TO ELIMINATE DUPLICATE STORAGE AREAS. IF INPUT/OUTPUT IS PERFORMED IN SEVERAL SEGMENTS, SOME RECORD MANAGER COMMON BLOCKS MAY BE MULTIPLY DEFINED (E.G., AOB.RM OR Q8.IO.).

THE GLOBAL CARD FOR ROUTINES WITH IMBEDDED SPECIAL CHARACTERS IN THE NAME MUST USE \$ SIGN (E.G., MYMAIN GLOBAL \$AOB.RM\$, \$Q8.IO.\$).

WHEN RECORD MANAGER COMMON IS GLOBAL TO A ROOT SEGMENT, THE LOADER MAY DETECT ERRORS IN INITIALIZING. IF SO, AN INCLUDE CARD WILL BE REQUIRED TO MOVE THE RM ROUTINES TO THAT SEGMENT (E.G., MYMAIN INCLUDE \$INCOM=\$).

CONTINUED DIRECTIVES MUST GO ALL THE WAY THROUGH COLUMN 72, EVEN IF A WORD IS SPLIT. THE NEXT CARD HAS A COMMA (,) IN COLUMN 1 AS THE CONTINUATION SIGNAL, THEN THE DIRECTIVE IS COMPLETED STARTING IN COLUMN 2.

*** ERROR MESSAGES ***

IF THERE ARE ERRORS OF ANY TYPE, MESSAGES APPEAR IN THE DAYFILE (LISTED AT THE END OF EACH JOB). SOME SPECIFIC ERROR MESSAGES APPEAR AS DESCRIBED BELOW. SOME OTHER ERRORS ARE DISCUSSED ON PAGE 14-1.

COMPILER ERRORS ARE PRINTED WITH THE LISTING OF THE SOURCE CODE. FOR FORTRAN, SEE CCRM, CHAP 4, AND FTN, CHAP III-2. FOR COBOL, SEE CCRM, CHAP 5, AND COBOL, APPENDIX G.

OBJECT TIME ERRORS SUCH AS BAD DATA FORMAT ARE NOTED AT THE END OF THE OUTPUT AND "FATAL ERROR NN" APPEARS IN THE DAYFILE.

LOADER AND NOS/BE ERRORS

DIAGNOSTIC MESSAGES ARE PUT IN THE DAYFILE BY THE NOS/BE OPERATING SYSTEM WHENEVER IT DETECTS ERRORS. FOR EXAMPLE, ERRORS FOUND BY THE LOADER ARE IDENTIFIED BY "FATAL LOADER ERROR -" FOLLOWED BY ADDITIONAL EXPLANATORY DIAGNOSTICS. SEE LOADER, APPENDIX C; LOADER/S, APPENDIX C.

MODE ERRORS

MODE ERRORS ARE DETECTED BY THE CENTRAL PROCESSOR AND MAY RESULT FROM ANY TYPE OF PROGRAM.

- ERROR MODE 1 ADDRESS OUT OF RANGE (USUALLY SUBSCRIPT ERROR OR A REFERENCE TO A SUBROUTINE OR FUNCTION NOT LOADED). IF THE ADDRESS = 4XXXXX THEN THERE IS A MISSING SUBROUTINE OR FUNCTION. (XXXXX IS THE ADDRESS OF THE REFERENCE TO THE ROUTINE.)
- 2 OPERAND OUT OF RANGE - INFINITE OPERAND
- 4 INDEFINITE OPERAND (E.G., ZERO DIVIDED BY ZERO OR UNINITIALIZED CORE)
- 3, 5, 6, 7 ARE COMBINATIONS OF MODES 1, 2, 4
- 0 INDICATES ATTEMPT TO EXECUTE ILLEGAL INSTRUCTION (PROBABLY DATA HAS OVERWRITTEN PART OF CODE.)

CERTAIN SYSTEM ERRORS RELATED TO FILE MANIPULATIONS (E.G. INPUT AND OUTPUT) GENERATE RECORD MANAGER DAYFILE MESSAGES

* RM ERROR NNN ON LFN XXXXXXXX *.

SEE RM, CHAPTER 6.

ALL FATAL ERRORS WILL CAUSE A SHORT DIAGNOSTIC DUMP (DMPX) TO BE PRINTED AT THE END OF THE USERS OUTPUT (CCRM, 4-15; NOSBE, 9-1; SCOPE, 9-1).

TAPE PARITY ERRORS

WHEN TAPE PARITY ERRORS ARE RECOVERED BY THE SYSTEM, THE MESSAGE CONTAINS "RVD". UNRECOVERED ERRORS PRINT "ERR" AND USUALLY ABORT JOB.

***** FILES AND TAPES *****

*** INTRODUCTION ***

THE 6000 SYSTEM IS A FILE-ORIENTED SYSTEM. A FILE MAY BE ACTUALLY A MAGNETIC TAPE, A DISK AREA, A DISK PACK, OR ANY OTHER SUCH DEVICE.

EVERY SEQUENTIAL FILE IS TREATED AS THOUGH IT WERE ASSIGNED TO A MAGNETIC TAPE. RANDOM ACCESS FILES MUST BE ON MASS STORAGE DEVICES.

ASSIGNMENT OF FILES TO ACTUAL DEVICES IS ACCOMPLISHED OUTSIDE THE PROGRAM WITH CONTROL CARDS. A FILE NEED NOT BE ASSIGNED TO THE SAME DEVICE OR EVEN TO THE SAME TYPE OF DEVICE ON DIFFERENT RUNS OF THE SAME PROGRAM. IF A FILE IS NOT ASSIGNED TO A PARTICULAR DEVICE WITH CONTROL CARDS, IT IS AUTOMATICALLY ASSIGNED TO A DISK AREA. IT IS RECOMMENDED THAT FOR PURELY SCRATCH FILES, DISK FILES BE USED. PLEASE PLACE FREQUENTLY USED FILES ON DISK.

INFORMATION FROM AN UNBLOCKED STRANGER TAPE TO BE STORED ON A PERMANENT FILE OR USED AS INPUT TO A COMPILER, MUST BE COPIED TO A SYSTEM STANDARD (SI) FILE. CARD AND PRINT LINE IMAGES AS WELL AS BLOCKED STRANGER TAPES MAY BE CONVERTED USING 'COPYRM' (CCRM, 6-6) OR CATALOGUED PROCEDURE 'COPYBLK' (CCLIB/P) WHICH UTILIZES FORM.

*** RECORD MANAGER ***

RECORD MANAGER (RM) IS A UNIFIED INPUT/OUTPUT PACKAGE USED BY MAJOR PRODUCT SETS. COBOL AND COMPASS MAY DEFINE MANY FILE TYPES WITHIN THE LANGUAGE. FTM UTILIZES SUCH FILES BY MEANS OF FILE DEFINITION CARDS.

RM SUPPORTS FIVE (5) TYPES OF FILE ORGANIZATIONS: SEQUENTIAL (SQ), INDEXED SEQUENTIAL (IS), WORD ADDRESSABLE (WA), DIRECT ADDRESS (DA), AND ACTUAL KEY (AK). (SEE RMFTN, RMCDB, RMUG)

ALL RM ROUTINES SHOULD BE IN THE 0.0 OVERLAY OF OVERLAY JOBS.

*** FILE ORGANIZER AND RECORD MANAGER (FORM) ***

THE FORM UTILITY CAN MANIPULATE RECORDS AND REORGANIZE FILES IN MANY WAYS. RECORDS MAY BE REBLOCKED FROM STRANGER FILES. TEST DATA MAY BE GENERATED BY SELECTING RECORDS FROM A LARGER INPUT FILE. FORM REPLACES THE SCOPE 3.3 INDEXED SEQUENTIAL UTILITIES FOR FILE MAINTENANCE. ALL INPUT/OUTPUT IS HANDLED BY RECORD MANAGER. USER MUST SUPPLY PROPER FILE CARDS AND DIRECTIVES. (SEE FORM REFERENCE MANUAL)

*** CHARACTERISTICS OF MASS STORAGE DEVICES ***

TWO TYPES OF ON-LINE ROTATING MASS STORAGE ARE AVAILABLE ON CERTAIN OF THE DTNSRDC CDC 6000 COMPUTERS: 844 AND 6638 DISKS. EACH 844 REMOVABLE DISK PACK HOLDS OVER 115 MILLION CHARACTERS. ON 844 DRIVES POSITIONING MAY TAKE PLACE ON ONE DRIVE WHILE READING OR WRITING ANOTHER (SEEK OVERLAP). SOME OF THE DEVICES ARE ASSIGNED FOR PERMANENT FILE STORAGE. SOME OF THE REPLACEABLE PACKS ARE AVAILABLE AS USER DEVICE SETS (UP TO 115 MILLION CHARACTERS PER JOB). THE NOS/BE OPERATING SYSTEM, QUEUES AND SCRATCH FILES RESIDE ON 844 PACKS. THESE PACKS HAVE SINGLE ERROR RECOVERY AND ARE VERY RELIABLE.

IT IS THE USERS RESPONSIBILITY TO DEFINE THE FILE STRUCTURE DESIRED. THE OPERATING SYSTEM ALLOCATES SPACE AS NEEDED IN UNITS OF RECORD BLOCKS (RB). DATA IS WRITTEN INTO PHYSICAL RECORD UNITS (PRU) WHEN A FILE IS GENERATED, UNTIL ALL THE PRU'S IN THE RB ARE FILLED, THEN ANOTHER RB IS ALLOCATED. ALTHOUGH PF STORAGE CHARGES ARE FOR PRU'S, ANY NUMBER OF PRU'S UP TO 56 USES UP A WHOLE RB OF THE AVAILABLE SYSTEM SPACE. SINCE MORE TIME IS SPENT IN POSITIONING THAN TRANSMITTING DATA, A MEMORY REQUEST STACK CONTAINS ALL OUTSTANDING MASS STORAGE FILE ACTION REQUESTS AND THE LEAST POSITIONING OPERATION IS PERFORMED FIRST. EACH 844 DISK CONTAINS 3232 RB'S WITH 56 PRU'S PER RB.

FORTRAN AND COBOL PROCESS ALL READ/WRITE OPERATIONS THROUGH BUFFERS. THE PROGRAMS MANIPULATE USER DEFINED RECORDS SUCH AS CODED UNIT RECORDS OR LOGICAL BINARY RECORDS. THE NOS/BE SYSTEM MANIPULATES DISK PRU'S OF 64 WORDS. TRAILING BLANKS ARE STRIPPED FROM CODED RECORDS AND A LINE TERMINATOR (12 BITS OF BINARY ZERO) IS ADDED BEFORE A RECORD IS PLACED IN A BUFFER FOR UNIT RECORD DEVICES.

SEVERAL RANDOM ACCESS TECHNIQUES ARE AVAILABLE FOR DATA FILES WHICH ARE NOT SEQUENTIAL IN NATURE. NOTE THAT ACCESSING MANY SMALL PIECES OF DATA IS TIME CONSUMING AND IT MAY BE BETTER TO READ SEQUENTIALLY, SINCE BOTH COBOL AND FORTRAN BUFFER I/O. JOB THROUGHPUT MAY BE DECREASED BY POOR CHOICE OF RECORD AND BLOCK STRUCTURE.

JUNE 1977

PAGE 3-3

*** PERMANENT FILES ***

PERMANENT FILES ARE DATA OR PROGRAM FILES PERMANENTLY STORED ON SELECTED DISK DEVICES AND ACCESSED BY THE NOS/BE COMMANDS GIVEN BELOW. IT IS THE ORIGINATOR'S RESPONSIBILITY TO PURGE A FILE WHEN IT IS NO LONGER NEEDED.

THERE ARE TWO (2) SEPARATE SETS OF PERMANENT FILES: ONE IS SHARED BY THE 6700 AND 6600; THE OTHER IS ON THE 6400.

ONLY UNCLASSIFIED PROGRAMS (SOURCE OR OBJECT) AND DATA MAY BE CATALOGED INTO PERMANENT FILES. NEW FILES WILL BE DUMPED TO BACKUP TAPE DAILY AND HELD ONE WEEK. THE TOTAL PERMANENT FILE BASE IS DUMPED ONCE A WEEK AND RETAINED ABOUT 3 WEEKS.

OCCASIONALLY SOME PERMANENT FILES BECOME UNREADABLE AND NOT ALL FILES MAY BE RECOVERABLE. THE USER SHOULD HAVE PROVISION FOR RECREATING THE FILES IF THIS SITUATION OCCURS. (SEE CCLIB/P: SELDUMP/SELLOAD)

PERMANENT FILE CONTROL CARDS MAY BE CONTINUED TO A SECOND CARD BY FILLING THE FIRST CARD THROUGH COLUMN 80 (OR STOPPING AFTER A COMMA) AND STARTING THE SECOND CARD IN COLUMN 1.

*** AUTOMATIC PERMANENT FILE PURGE ***

PERMANENT FILES MAY BE PURGED BY THE COMPUTER CENTER FOR ONE OF THE FOLLOWING REASONS:

- 1) INVALID JOB ORDER NUMBER FOR AC
OR CANCELLED JOB ORDER NUMBER
- 2) INVALID ID (E.G., NOT USER INITIALS)
- 3) NOT ACCESSED WITHIN THE LAST 30 DAYS (6700/6600)
OR 15 DAYS (6400)

THE NON-ACCESSED FILES WHICH ARE PURGED WILL BE KEPT ON TAPE FOR AT LEAST ONE (1) MONTH. ANYONE WISHING TO RECOVER SUCH A PURGED FILE SHOULD CALL THE PERMANENT FILE LIBRARIAN, CODE 1892.1, (202) 227-1907. A NOMINAL FEE WILL BE CHARGED FOR THIS SERVICE.

NOTE THAT AN UNNEEDED FILE SHOULD BE PURGED BY THE USER RATHER THAN WAITING OVER 30 DAYS FOR THE AUTOMATIC PURGE, SINCE THERE IS A CHARGE FOR PERMANENT FILE STORAGE AS LONG AS THE FILE EXISTS AS A PERMANENT FILE IN THE SYSTEM.

THE EXPIRATION DATE IS NOT PRESENTLY USED FOR FILE PURGING. HOWEVER, THE USER MAY USE THIS PARAMETER AS A REMINDER OF THE INTENDED LIFE EXPECTANCY OF A FILE. THE DEFAULT EXPIRATION DATE IS THIRTY (30) DAYS AFTER CREATION. INFINITE RETENTION IS RESERVED FOR SYSTEM FILES.

A FILE IS LIMITED TO NO MORE THAN 5000 PRU'S ON THE 6700/6600, 3000 PRU'S ON THE 6400, EXCEPT BY SPECIAL WRITTEN PERMISSION OF CODE 1892.

JUNE 1977

PAGE 3-4

*** USER AUDIT ***

IN ORDER TO DETERMINE WHAT FILES A USER HAS ON THE PERMANENT FILE STORAGE, BOTH STATIC AND DYNAMIC AUDIT PROGRAMS ARE PROVIDED. THE OUTPUT LIST OF THE DYNAMIC AUDIT IS OF THE PERMANENT FILES EXISTING AT THE MOMENT THE AUDIT PROGRAM IS RUN. THE STATIC AUDIT LISTS THE FILES AS THEY EXISTED (USUALLY) AT THE END OF THE PREVIOUS DAY. (SEE GCRM, 3-12, FOR USER DEVICE SET AUDIT.)

STATIC AUDIT

BEGIN,AUDIT,TYPE=STATIC,<PARAMETERS>.

DYNAMIC AUDIT

AUDIT,<PARAMETERS>. (UNSORTED)

BEGIN,AUDIT,<PARAMETERS>. (SORTED)

AUDIT PARAMETERS

ALL AUDIT PARAMETERS ARE OPTIONAL AND ARE KEYWORDS. ONLY THE LAST TWO DIGITS OF THE ACCOUNT NUMBER WILL APPEAR ON AN AUDIT.

LF= SPECIFY THE FILE ON WHICH THE AUDIT INFORMATION IS TO BE WRITTEN.
(DEFAULT: LF=OUTPUT)

AI= TYPE OF AUDIT DESIRED

AI=F -- FULL AUDIT (2 LINES PER FILE)

(PRUS, ID, CY, FULL PFN, AC SEGMENT, CREATION DATE,
DATE/TIME LAST ATTACHED, DATE/TIME LAST ALTERED, # USES,
RBS, # ALTERS/REWRITES, # EXTENDS, EXPIRATION DATE,
DISK VSN)

AI=I -- INTERCOM AUDIT (1 72-COLUMN LINE PER FILE)

(ID, PRUS, CY, 29-CHARACTERS OF PFN, AC SEGMENT, CREATION
DATE, DATE LAST ATTACHED, # USES)
(DEFAULT FOR INTERCOM STATIC AND UNSORTED DYNAMIC)

AI=P -- PARTIAL AUDIT (1 LINE PER FILE)

(PRUS, ID, CY, FULL PFN, AC SEGMENT, CREATION DATE, DATE
LAST ATTACHED, DATE LAST ALTERED, # USES)
(DEFAULT FOR OTHER AUDITS)

AI=S -- SHORT AUDIT (ID, PRUS, CY, FULL PFN)

AC ACCOUNT NUMBER TO BE AUDITED

(ONLY THE SEGMENT (LAST 2 DIGITS) ARE LISTED)

AC -- AUDIT BY ACCOUNT NUMBER ON CHARGE CARD (BATCH)
OR LOGIN (INTERCOM) (NOT FOR BEGIN,AUDIT,....)

AC=JJJJJJJJJJ -- AUDIT FILES BY 10-DIGIT ACCOUNT NUMBER
(MUST MATCH CHARGE CARD OR LOGIN)

AC=JJJJJJJJJ00 -- AUDIT BY FIRST 8 DIGITS OF ACCOUNT NUMBER
(MUST MATCH CHARGE CARD OR LOGIN)

ID= AUDIT FILES BY USER ID XXXX (DEFAULT: FROM CHARGE CARD/LOGIN)

IF BOTH ID AND AC ARE SPECIFIED, THAT SUBSET WILL BE LISTED.

WHEN NO PARAMETERS ARE SPECIFIED (AUDIT.), THE DEFAULTS USED ARE

LF=OUTPUT,AI=I,ID=XXXX INTERCOM STATIC AND UNSORTED DYNAMIC AUDITS

LF=OUTPUT,AI=P,ID=XXXX ALL OTHER AUDITS

JUNE 1977

PAGE 3-5

*** PERMANENT FILE COMMANDS ***

REQUEST,LFN,*PF.

MUST PRECEDE THE INITIAL WRITING OF FILE LFN.

CATALOG,LFN,PFN,ID=XXXX,AC=JJJJJJJJJJ,<PARAMETERS>.

CATALOG,LFN,ID=XXXX,AC=JJJJJJJJJJ,<PARAMETERS>.

ENTER THE FILE IN PERMANENT FILE DIRECTORY AFTER WRITING

ATTACH,LFN,PFN,ID=XXXX,<PARAMETERS>.

ATTACH,PFN,ID=XXXX,<PARAMETERS>.

TO USE A PREVIOUSLY CATALOGED FILE.

PURGE,LFN,PFN,ID=XXXX,<PARAMETERS>.

PURGE,PFN,ID=XXXX,<PARAMETERS>.

IF FILE TO BE PURGED HAS BEEN ATTACHED WITH CONTROL
PERMISSION PREVIOUSLY IN THE JOB, ONLY LFN IS REQUIRED.

RENAME,LFN,PFN,<PARAMETERS>.

CHANGES NAME OR ALTERS PASSWORDS, CYCLE, ID, OR AC.
HOWEVER, FILE MUST BE ATTACHED WITH ALL PERMISSIONS BEFORE
RENAME. CANNOT CHANGE PFN, ID OR PW FOR ONE CYCLE
OF A MULTI-CYCLE FILE. CANNOT RENAME IF MR=1.

EXTEND,LFN.

ADD PERMANENT DATA TO END OF ATTACHED PERMANENT FILE.

ALTER,LFN.

CHANGE LENGTH OF ATTACHED PERMANENT FILE.

REQUIRED PARAMETERS

LFN LOGICAL FILE NAME, 1 TO 7 CHARACTERS, FIRST MUST BE ALPHABETIC.

PFN PERMANENT FILE NAME FOR SYSTEM CATALOG, UP TO 40 CHARACTERS.

IF LFN IS NOT SPECIFIED, FIRST 7 CHARACTERS OF PFN ARE LFN.

IF PFN NOT SPECIFIED, LFN IS ALSO PFN.

ID=XXXX REGISTERED INITIALS OF FILE OWNER (AS ON CHARGE CARD).

REQUIRED ON CATALOG, ATTACH, AND PURGE (LONG FORM).

SEE NEXT PAGE FOR OPTIONAL PARAMETERS.

PARAMETERS (USUALLY OPTIONAL)

ALL OPTIONAL PERMANENT FILE PARAMETERS ARE KEYWORDS. A KEYWORD IS FOLLOWED BY '=', THEN THE VALUE WITH NO BLANKS.

AC=JJJJJJJJJJ JOB NUMBER TO CHARGE STORAGE OF FILE. WHEN OMITTED OR INVALID FORMAT (E.G., TOO FEW CHARACTERS), AC IS TAKEN FROM CHARGE CARD OR LOGIN.

CY= CYCLE NUMBER 1 TO 999 (INITIAL CREATE DEFAULT IS 1) ONLY FIVE (5) CYCLES MAY EXIST AT ANY ONE TIME. NOTE THAT THE HIGHEST NUMBERED CYCLE WILL BE ATTACHED OR PURGED IF CY AND LC ARE NOT SPECIFIED.

LC=1 ACCESS THE LOWEST NUMBERED CYCLE ON ATTACH OR PURGE.

MR=1 ALLOWS MULTI-READ ACCESS ON ATTACH (SIMULTANEOUS USE BY DIFFERENT JOBS). ALSO, JOB MAY BE RERUN BY SYSTEM IF ERROR. A FILE ATTACHED WITH READ-ONLY PERMISSION CANNOT BE PURGED, RENAMED OR EXTENDED.

RP= RETENTION PERIOD IN DAYS, UP TO 720. (DEFAULT: 30)

TK= TURNKEY PASSWORD

RD= READ PASSWORD

XR= PASSWORD DEFINITION FOR READ-ONLY FILES. SETS CN, MD, EX. USED ON CATALOG INSTEAD OF MR=1 ON ATTACH.

EX= EXTEND PASSWORD

MD= MODIFY PASSWORD

CN= CONTROL PASSWORD

NOTE: THE ABOVE PASSWORDS ARE 1-9 ALPHAMERIC CHARACTERS AND ARE USED ONLY TO DEFINE PASSWORDS ON CATALOG OR RENAME.

PW=LIST USED TO SUBMIT PASSWORDS. A MAXIMUM OF FIVE (5) PASSWORDS FOR ATTACH OR FOR CATALOG OF A NEW CYCLE.

*** HINTS ON PERMANENT FILE USAGE ***

1. ATTEMPT TO PURGE A NON-EXISTING FILE WILL BE FLAGGED BUT NON-FATAL.
2. CATALOG WITHOUT CY= PARAMETER FOR PRE-EXISTING FILE WILL CATALOG THE NEXT HIGHER CYCLE IF CONTROL PERMISSION IS GRANTED.
3. FILES SHOULD BE CATALOGED WITH XR AND PW PASSWORDS TO TAKE FULL ADVANTAGE OF MULTIPLE READ ON ATTACH AND MULTIPLE CYCLES ON CATALOG.
4. SYSTEM PERMANENT FILES WILL HAVE PUBLIC ACCESS ID AND MAY BE ATTACHED BY PFN ALONE.

EXAMPLES

CATALOG,LGO,SPSOBJ,ID=XXXX,AC=1111111111,XR=READ,PW=READ.

(IDENTICAL CATALOG WILL CREATE AN ADDITIONAL CYCLE)

ATTACH,SPSOBJ,ID=XXXX.

ONLY READ PERMISSION, LFN IS PFN

PURGE,OLD,SPSOBJ,ID=XXXX,PW=READ,LC=1.

DISCARD LOWEST CY

ATTACH,SC4020.

A SYSTEM LIBRARY, ID NOT REQUIRED

CATALOG,ANYCPL,ID=XXXX,XR=RD,PW=RD.

JOB AC USED

*** PERMANENT FILE USER RETURN CODES ***

PERMANENT FILE ERRORS ARE REFERENCED BY THE ERROR NUMBER. INTERCOM REFERENCES ARE DECIMAL CODES.

| DECIMAL | OCIAL | MEANING | RELATED COMMANDS |
|---------|---------|---|--------------------------------|
| 0 | 000 | FUNCTION SUCCESSFUL | |
| 1 | 001 | PFN/ID ERROR | |
| 2 | 002 | LFN ALREADY IN USE | ATTACH |
| 3 | 003 | UNKNOWN LFN | CATALOG, EXTEND, PURGE, RENAME |
| 4 | 004 | NO ROOM FOR EXTRA CYCLE (5 MAX) | CATALOG |
| 5 | 005 | PF CATALOG FULL | CATALOG, EXTEND |
| 6 | 006 | NO LFN OR PFN | |
| 7 | 007 | (NOT USED) | |
| 8 | 010 | LATEST INDEX NOT WRITTEN FOR A RANDOM FILE | CATALOG, EXTEND |
| 9 | 011 | FILE NOT ON A PF DEVICE | CATALOG |
| 10 | 012 | FILE NOT CATALOGED, SN=XXXXXXX | ATTACH |
| 11 | 013 | ARCHIVE RETRIEVAL ABORTED | ATTACH |
| 12 | 014 | (NOT USED) | |
| 13 | 015 | CY # LIMIT REACHED (999) | CATALOG |
| 14 | 016 | PF DIRECTORY FULL | CATALOG |
| 15 | 017 | FUNCTION ATTEMPTED ON NON-PERMANENT FILE | CATALOG, EXTEND, PURGE, RENAME |
| 16 | 020 | FUNCTION ATTEMPTED ON NON-LOCAL FILE | |
| 17 | 021 | (NOT USED) | |
| 18 | 022 | FILE NEVER ASSIGNED TO A DEVICE (FILE NEVER WRITTEN ON) | CATALOG |
| 19 | 023 | CYCLE INCOMPLETE | ATTACH |
| 20 | 024 | PF ALREADY ATTACHED | ATTACH |
| 21 | 025 | FILE UNAVAILABLE | ATTACH |
| 22 | 026 | (NOT USED) | |
| 23 | 027 | ILLEGAL LFN | |
| 24 | 030 | FILE DUMPED | ATTACH |
| 25 | 031 | ILLEGAL FUNCTION CODE | |
| 26 | 032 | (NOT USED) | |
| 27 | 033 | ALTER NEEDS EXCLUSIVE ACCESS | ALTER |
| 28 | 034 | (NOT USED) | |
| 29 | 035 | FILE ALREADY IN SYSTEM | CATALOG |
| 30 | 036 | (NOT USED) | |
| 31 | 037 | (NOT USED) | |
| 32 | 040 | ILLEGAL SETNAME SPECIFIED | |
| 33 | 041 | DEVICE SET NOT MOUNTED AT CONTROL POINT | |
| 34 | 042 | RBT CHAIN TOO LARGE FOR PFC | |
| 35 | 043 | FILE RESIDES ON UNAVAILABLE DEVICE | |
| 36-55 | 044-067 | (NOT USED) | |
| 56 | 070 | PFM STOPPED BY SYSTEM | |
| 57 | 071 | INCORRECT PERMISSION | |
| 58 | 072 | FDB ADDRESS INVALID | |
| 59 | 073 | I/O ERROR ON PFD/PFC WRITE | |
| 60 | 074 | ST PARAMETER ILLEGAL WITH USER DEVICE SET | |

PAGE 3-8

CONSTRUCT PERMANENT FILE FROM CARDS

USE OF PERMANENT SOURCE FILE

PURGE OBSOLETE FILE

```
JOB CARD  
CHARGE CARD  
PURGE,(NOGOOD,OLDFILE,ID=XXXX,PW=RESERVE) ** ALLOW CONTROL  
PERMISSION  
.. 6/7/8/9 EOF
```


CATALOG NEW CYCLE OF EXISTING FILE

```

JOB CARD
CHARGE CARD
REQUEST,NEWCYC,*PF.
ATTACH,PROG,PROGRAM,ID=CASG.          ** NO MULTIPLE READ WHILE MODIFY
COPYR(INPUT,NEWCYC)
COPYF(PROG,NEWCYC)                    ** ADD OLD INFO TO NEWCYC
CATALOG,NEWCYC,PROGRAM,ID=XXXX,PW=RESERVE.
                                         ** WILL ASSIGN NEXT CYCLE NUMBER
                                         (UP TO 999)
'   7/8/9      EOR
    (NEW CARDS TO BE PUT INTO NEXT CYCLE)
"   6/7/8/9    EOF

```

REPLACE JOB NUMBER, ADD PERMISSIONS

```

JOB CARD
CHARGE CARD
ATTACH,PROG,PROGRAM,ID=XXXX,PW=RESERVE.          ** CONTROL PERMISSION
COMMENT.    NOTE MULTIPLE READ MUST NOT BE ALLOWED.
RENAME,PROG,AC=NEWJOBNNBR,EX=RESERVE,MD=RESERVE.
COMMENT.    PF NAME STAYS THE SAME - ALL CYCLES
COMMENT.    NEW JOB ORDER NUMBER - THIS CYCLE
COMMENT.    ADDITIONAL PASSWORDS - ALL CYCLES
"   6/7/8/9    EOF

```

EXTEND EXISTING FILE

INFORMATION MAY BE ADDED TO AN EXISTING PERMANENT FILE ONLY AT THE END-OF-FILE (EOI). THIS EXAMPLE ILLUSTRATES ADDING ONE RECORD TO A FILE. 999 IN THE FIRST COPYR SHOULD BE SUFFICIENT TO BYPASS ALL EXISTING RECORDS. IF AN END-OF-FILE EXISTS, THE SECOND COPYR WILL ABORT SINCE THE FILE IS NOT AT EOI. THEREFORE, THE ALTER IS NEEDED TO REPOSITION THE EOI (ANY INFORMATION PAST THIS POINT IN THE ORIGINAL FILE IS LOST). NOTE THAT ALTER WILL PUT 2 LINES BEGINNING WITH 'EX' INTO THE DAYFILE.

```

.   OTHER CONTROL CARDS AS REQUIRED
.
.
ATTACH,DATA,ID=XXXX,PW=LARGE.          ** EXTEND PERMISSION
COMMENT.    NOTE MULTIPLE READ MUST NOT BE ALLOWED.
COPYR,DATA,TEMP,999.                  ** POSITION TO END OF DATA
ALTER,DATA.                            ** REPOSITION EOI
COPYR,INPUT,DATA.                      ** ADD NEW RECORD
EXTEND,DATA.                           ** REPOSITION EOI
'   7/8/9      EOR
    (NEW DATA RECORD TO BE ADDED)
"   6/7/8/9    EOF

```

*** DEVICE SETS ***

ON DYNASDC CDC 6000 COMPUTERS SOME OF THE REMOVABLE DISK PACKS ARE AVAILABLE FOR USERS. DEVICE SETS (DS) UNDER NOS/BE 1.0 HAVE SEVERAL ADVANTAGES OVER THE PRIVATE DISK PACKS FORMERLY USED ON SCOPE 3.3. THE OWNER OF A DS HAS ALL CAPABILITIES NECESSARY TO MANAGE HIS OWN PERMANENT FILE SYSTEM. USER FILES MAY BE WRITTEN TO A DS AND MADE PERMANENT USING STANDARD PERMANENT FILE CONTROL CARDS WITH ONE ADDITIONAL PARAMETER. A DS MAY BE USED BY SEVERAL JOBS SIMULTANEOUSLY. JOBS ON BOTH THE 6600 AND 6700 MAY ACCESS A GIVEN DS SIMULTANEOUSLY. THE STANDARD AUDIT CONTROL CARD CONTAINING ONE ADDITIONAL PARAMETER IS USED TO LIST THE CONTENTS OF THE PACK. ALL AUDIT OPTIONS ARE AVAILABLE. EACH PACK HAS A UNIQUE VOLUME SERIAL NUMBER (VSN).

DEVICE SETS SHOULD NOT BE USED FOR ON-LINE STORAGE (USE PERMANENT FILES INSTEAD). NOS/BE 1.0 TREATS DEVICE SETS THE SAME AS MAGNETIC TAPES. THERE ARE A LIMITED NUMBER OF DRIVES FOR DEVICE SETS. JOBS REQUIRING THEM MUST WAIT UNTIL THE REQUIRED DRIVE(S) ARE AVAILABLE.

IT IS THE USER'S RESPONSIBILITY TO PROVIDE BACKUP (TAPE OR DISK) FOR DEVICE SETS. THE COMPUTER CENTER PROVIDES PERIODIC BACKUP FOR FILES ON PUBLIC DEVICES ONLY.

DEVICE SET PACKS ARE ASSIGNED BY THE TAPE LIBRARIAN, CODE 1896, (202) 227-1227. WITH INFORMATION PROVIDED BY THE OWNER, USER SERVICES MUST INITIALIZE THE PACK WITH LABEL, WRITE SYSTEM FLAW MESSAGES SO THAT DATA WILL NOT CAUSE UNCORRECTABLE RMS ERRORS, AND CREATE ON THE DS A SPECIAL PERMANENT FILE 'DUM' TO ALLOW BACKUP DUMP CAPABILITY. INFORMATION WHICH MUST BE PROVIDED INCLUDES:

MASTER PACK (MP) VSN, USUALLY SAME AS VOLSER
VOLSER THE UNIQUE SIX CHARACTER VSN
SET NAME (SN) 1-7 ALPHANUMERICS STARTING WITH A LETTER
RETENTION PERIOD FOR THE PACK (RP) UP TO 720 DAYS
UNIQUE RD, MD, AND CN PASSWORDS FOR THE 'DUM' FILE AS REQUIRED FOR
MODE 1, 2 OR 3 TAPE BACKUP DUMPS
JOB ORDER NUMBER (AC) FOR CATALOG OF 'DUM'
SEE NOSBE, CHAPTER 4; SCOPE, 4-22 TO 4-25.

*** DEVICE SET PASSWORD ***

AN ADDITIONAL LEVEL OF SECURITY FOR DEVICE SETS IS AVAILABLE AS A PASSWORD ON THE MOUNT COMMAND. TO ACTIVATE THIS FEATURE, THE USER MUST CATALOG ON THE PACK A FILE WITH PFN AND ID OF "PASSWORD" WITH A 1-9 CHARACTER TURNKEY PASSWORD (TK).

FOR EXAMPLE:

REQUEST,PASS,*PF,SN=SETNAME.

REWIND,PASS.

CATALOG,PASS,PASSWORD,ID=PASSWORD,TK=XXXX.

EACH MOUNT CARD FOR THAT PACK MUST THEN INCLUDE THE PW PARAMETER

MOUNT,VSN=VOLSER,SN=SETNAME,PW=XXXX.

THE DEVICE SET OWNER MAY CHANGE THIS PASSWORD BY RENAMING THE PASSWORD FILE WITH A DIFFERENT TURNKEY.

DEVICE SET CONTROL CARDS

PAUSE. JOB REQUIRES DISK PACK VOLSER.
NOTIFY OPERATOR TO PREPARE REQUIRED PACK FOR MOUNTING.

MOUNT,VSN=VOLSER,SN=SETNAME.
MOUNT,VSN=VOLSER,SN=SETNAME,PW=USERPASS.
REQUEST AN EXISTING DS TO BE PHYSICALLY MOUNTED (IF NOT
ALREADY MOUNTED) AND LOGICALLY ASSOCIATED WITH THIS JOB.
MOUNT CARD MUST PRECEDE ALL OTHER DS CONTROL CARDS EXCEPT
RECOVER (SEE CCRM, 3-12).

DSMCUNT,VSN=VOLSER,SN=SETNAME.
LOGICALLY DISASSOCIATE THE DS FROM THIS JOB. (AUTOMATIC
AT END-OF-JOB.) RETURNS ALL FILES ATTACHED FROM THE PACK.

SETNAME,SETNAME.
ESTABLISH A DEFAULT SN FOR FOLLOWING ATTACH AND REQUEST
CARDS. USE ONLY WHEN ALL FILES TO BE ATTACHED RESIDE ON
THE PACK.

SETNAME.
RE-ESTABLISHES THE SYSTEM DEVICE AS DEFAULT FOR ATTACH AND
REQUEST.

REQUEST,LFN,*PF,SN=SETNAME.
REQUEST,LFN,*PF,SN. (IF PRECEDED BY SETNAME CARD)
REQUEST FILE LFN BE WRITTEN ON THE DS.

ATTACH,LFN,ID=XXXX,SN=SETNAME,....
ATTACH,LFN,ID=XXXX. (IF PRECEDED BY SETNAME CARD)
ATTACH THE INDICATED PERMANENT FILE FROM THE DS.

CATALOG CARDS AND OTHER ATTACH PARAMETERS ARE UNCHANGED FROM SYSTEM
PERMANENT FILE DEVICES. FILES ARE REMOVED FROM THE PACK BY PURGE.

DEVICE SET UTILITIES

AUDIT, ID=XXXX, SN=SETNAME.

AUDIT THE FILES OF XXXX ON DS 'SETNAME'.
SEE CORM 3-4 FOR OTHER AUDIT PARAMETERS.

DUMPF (PW=RDPASSW, SN=SETNAME)

CREATE BACKUP TAPE DUMP OF ALL FILES ON DS.
SEE EXAMPLE FOR CORRECT SEQUENCE OF INSTRUCTIONS.

LOADPF (SN=SETNAME)

RELOAD ALL FILES FROM TAPE TO SET.

LOADPF (I, SN=SETNAME)

ALLOWS FOR SELECTIVE LOAD OF FILES, USING DATA CARDS
OF THE FORM: PF=PFN1, ID=XXXX.

RECOVER, VSN=VOLSER, SN=SETNAME.

IF A MACHINE FAILURE OCCURS WHILE A DS IS MOUNTED, THIS UTILITY
(NOT PRECEDED BY A MOUNT) VALIDATES THE DS, VERIFIES AND
RECREATES ALL CRITICAL DISK TABLES. (REQUIRES PFL, 50000.)

TRANSPF

UTILITY TO BACKUP ONE DEVICE SET ONTO ANOTHER PACK. SEE
CCLIB/P: TRANPAK.

BACKUP DUMP

```

JOB, MT1, RP1, T500.                                USER NAME   CODE
CHARGE, XXXX, JJJJJJJJJJ.
PAUSE. JOB REQUIRES DISK PACK DV9999.
MOUNT, VSN=DV9999, SN=SETNAME.
VSN, DUMTAPE=CA8888/CA8889.                        DUMTAPE AND L= MUST BE AS SHOWN
LABEL (DUMTAPE, L=$DUMPF TAPE- NEW$, F=S, W, D=HY, RING)
DUMPF (PW=RDPASSW, SN=SETNAME)
" 7/8/9      EOR
" 6/7/8/9    EOF

```

SELECTIVE RELOAD

```

JOB, MT1, RP1.                                USER NAME   CODE
CHARGE, XXXX, JJJJJJJJJJ.
PAUSE. JOB REQUIRES DISK PACK DV9999.
MOUNT, VSN=DV9999, SN=SETNAME.
VSN, DUMTAPE=CA8888/CA8889.                        DUMTAPE AND L= MUST BE AS SHOWN
LABEL (DUMTAPE, L=$DUMPF TAPE- NEW$, F=S, R, D=HY, NORING)
LOADPF (I, SN=SETNAME)
" 7/8/9      EOR
PF=MYFILE1, ID=XXXX.
PF=MYFILE2, ID=XXXX, CY=2.
" 6/7/8/9    EOF

```

*** TAPE CHARACTERISTICS ***

TAPE DRIVES ON THE 6400 ARE ON ONE CHANNEL, THUS THERE ARE NO CONCURRENT READ OR WRITE OPERATIONS. THERE ARE TWO CHANNELS FOR TAPE DRIVES ON THE 6700 AND 6600.

SYSTEM STANDARD (SI) TAPES SHOULD BE LABELLED. BLOCKING HAS FIXED MAXIMUM SIZES. SEVEN-TRACK TAPES MAY BE 200, 556, OR 800 BPI DENSITY. CODED INFORMATION (EVEN PARITY) IS WRITTEN IN 128 WORD PHYSICAL RECORD UNITS (PRU). FORTRAN CODED (UNIT) RECORDS HAVE DEFAULT MAXIMUM OF 150 CHARACTERS, WHICH ARE BUFFERED INTO THE 128 WORD PRU. BINARY (ODD PARITY) PRU SIZE IS 512 WORDS. FORTRAN BINARY LOGICAL RECORDS ARE BLOCKED INTO THE 512 WORD PHYSICAL RECORDS AUTOMATICALLY. FORTRAN BINARY LOGICAL RECORDS MAY EXCEED 512 WORDS.

CDC'S DESIGNATION OF STRANGER TAPES ARE THOSE GENERATED BY EQUIPMENT OTHER THAN CDC 6000 WHICH HAVE A MAXIMUM OF 512 WORD PHYSICAL RECORDS. FORTRAN MAY CREATE STRANGER TAPES FOR OTHER EQUIPMENT BY USING:

- A) BUFFER OUT STATEMENTS TO AVOID HAVING NOS/BE LEVEL NUMBERS;
- B) FORMATTED WRITE WITH TAPE BLOCKING CHARACTERISTICS DEFINED ON 'FILE' AND 'LDSET' CARDS (SEE CCRM, 3-18, 2-16).

TAPES WITH RECORDS LONGER THAN 512 WORDS MAY ONLY BE READ AS 'L' TAPES. A STRANGER TAPE OF UNBLOCKED CARD IMAGE OR PRINT IMAGE RECORDS MAY REQUIRE OVER ONE WALL CLOCK HOUR OF TIME TO READ. IF POSSIBLE, READ SUCH A TAPE ONLY ONCE ON CDC 6000, WHILE REBLOCKING TO A SI FILE. (SEE CCRM, 6-6; COPYRM; CCLIB/P; COPYBLK).

NINE-TRACK TAPES (ONLY AVAILABLE ON THE CDC 6700) ARE ALL ODD PARITY, WITH DENSITY OF 800 BPI. BINARY INFORMATION MAY BE PACKED TO USE ALL 8 CHANNELS ON THE TAPE. 'L' TAPE FORMAT IS NOT PERMITTED.

ALL INFORMATION CONCERNING THE PHYSICAL CHARACTERISTICS (MODE, TYPE, DENSITY) OF THE FILE IS SPECIFIED ON CONTROL CARDS. THE TOTAL NUMBER OF TAPES TO BE MOUNTED SIMULTANEOUSLY DURING THE JOB IS A REQUIRED PARAMETER ON THE JOB CARD. VSN AND LABEL (OR REQUEST) INFORMATION IS REQUIRED FOR EACH TAPE. THE LABEL CARD SHOULD BE PLACED JUST BEFORE LOADING THE PROGRAM WHICH REQUIRES THE TAPE. THUS, IF THE PREVIOUS STEP FAILS, THE JOB WILL ABORT WITHOUT CAUSING THE OPERATOR TO MOUNT THE TAPE. SCRATCH FILES SHOULD BE ON DISK, NOT ON TAPE.

*** TAPE ASSIGNMENT ***

TWO CLASSES OF TAPE STORAGE ARE PROVIDED IN THE COMPUTER CENTER. TAPES WHICH ARE SELDOM USED ON THE CDC 6000 SERIES COMPUTERS, WHICH WILL BE USED ON SEVERAL DIFFERENT SYSTEMS, OR WHICH ARE NORMALLY RETAINED BY THE OWNER ARE ASSIGNED A TEMPORARY SLOT NUMBER FOR UP TO A 24-HOUR WHICH PERIOD AT THE ADP CONTROL CENTER OF THE COMPUTER FOR THEY ARE INTENDED. AT THE END OF THE DAY'S PROCESSING (OR EARLIER AT THE USER'S REQUEST) THESE ARE RETURNED TO THE USER AND WILL REQUIRE A DIFFERENT SLOT NUMBER ASSIGNMENT FOR NEXT USE.

VRNO ON THE CDC 6000 IS 'SLOTXX=ID'
WHERE XX IS THE ASSIGNED SLOT
ID IS THE USER'S EXTERNAL STICKER ON THE TAPE REEL
(NOT EXCEEDING SIX (6) CHARACTERS).

TAPES WHICH ARE USED FREQUENTLY SHOULD BE PERMANENTLY STORED AT THE PROPER COMPUTER (I.E., 6700/6600 OR 6400) IN THE DIEBOLD CABINETS WHICH ARE LARGE ROTATING STORAGE FOR SEVERAL HUNDRED TAPES. THESE TAPES ARE ASSIGNED A PERMANENT EXTERNAL LABEL (AND VRNO) WITH FILE NUMBER, SHELF NUMBER, AND TAPE NUMBER SUCH AS 'CA0190'.

FOR ASSIGNMENT OF TAPE STORAGE LOCATIONS AND 6000 SERIES VISUAL REEL NUMBERS SEE THE TAPE LIBRARIAN, CODE 1896, (202) 227-1227.

*** TAPE USAGE AND CLEANING ***

TAPES SHOULD BE STORED IN CLOSED CONTAINERS IN RACKS WHICH GIVE THEM VERTICAL SUPPORT. TAPES MAY NOT BE SPLICED. THEY SHOULD BE READ AND REWOUND EVERY SIX MONTHS AT LEAST. LOGS SHOULD BE KEPT ON CREATION DATES OF TAPES. OLD TAPES MAY BE CONSIDERED FOR REPLACEMENT.

ALL TAPES USED ON CDC 6000 COMPUTERS SHOULD BE STANDARD LABELLED. STRANGER TAPES SHOULD BE READ ONCE AND COPIED TO SI TAPES.

IF SURFACE DIRT CAUSES PARITY ERRORS ON A TAPE, THE TAPE SHOULD BE CLEANED. THIS OFF-LINE PROCESS DOES NOT DESTROY THE INFORMATION ON THE TAPE. SUBMIT OFF-LINE WORK REQUEST TO THE TAPE LIBRARIAN. IF A TAPE RECEIVES HEAVY USAGE, CLEANING IT AFTER TEN OR MORE USES MAY REDUCE THE INCIDENCE OF PARITY ERRORS.

IF, AFTER A TAPE HAS BEEN CLEANED, IT STILL HAS MANY PARITY ERRORS, IT SHOULD BE EXCHANGED FOR A NEW BLANK TAPE. THE USER MUST BE SURE THE FILE(S) CAN BE RE-CREATED ON THE NEW TAPE.

A REQUEST TO TEST OR CERTIFY A TAPE WILL DESTROY ANY INFORMATION ON THE TAPE.

PLEASE NOTIFY GEORGE SMITH, CODE 1892.1, (202) 227-1907, OF ANY TAPE PROBLEMS.

*** VSN ***

ALL JOBS WHICH USE MAGNETIC TAPE, LABELLED OR UNLABELLED, MUST USE THE 'VSN' CONTROL CARD OR THE 'VSN=' PARAMETER ON THE LABEL OR REQUEST CARD. VSN CARDS MUST PRECEDE THE LABEL OR REQUEST CARD FOR THE FILE NAMED. A SINGLE CARD CAN CONTAIN DECLARATIONS FOR SEVERAL FILE NAMES. IF A LOGICAL FILE NAME IS REUSED BY A DIFFERENT REEL DURING ONE JOB, THE UNLOAD OR RETURN CARD MUST PRECEDE THE VSN CONTROL CARD FOR THE SECOND FILE. EACH VOLUME SERIAL NUMBER IS A MAXIMUM OF SIX (6) CHARACTERS. THE SLOT IDENTIFICATION ON VSN CARDS SHOULD PRECEDE THE EXTERNAL STICKER LABEL, SEPARATED BY '='. BE EXTREMELY CAREFUL OF 0 (ZERO) AND O (OH) CHARACTERS AS A COMPUTER PROGRAM CHECKS THIS LABEL. UNLABELLED TAPES REQUIRE OPERATOR INTERVENTION EACH TIME THEY ARE MOUNTED AND SHOULD BE AVOIDED.

| | |
|--|---|
| VSN(LFN1=VSN1,LFN2=VSN2...) | DIEBOLD TAPE |
| VSN(LFNM=VSN1/VSN2) | MULTIREEL FILE |
| VSN(LFNS=VSN1=VSN2) | SLOT TAPE OR OTHER REEL WHERE PHYSICAL STORAGE LABEL IS NOT SAME AS TAPE LABEL AT ORIGINAL WRITE |
| VSN(LFNS=VSN11=VSN12/VSN21=VSN22) MULTI-REEL SLOT TAPES | |
| LFN1 IS LOCAL FILE NAME ON A SUBSEQUENT LABEL OR REQUEST CONTROL CARD. | |
| VSNI IS VOLUME SERIAL NUMBER, E.G. CA0599 OR SLOT88. NO MORE THAN SIX CHARACTERS. | |

SLOT TAPES NEED THE ALTERNATE TAPE VOLUME SPECIFICATION. WHEN TWO OR MORE SIX-CHARACTER VSNI ARE EQUATED, THE COMPUTER WILL ALLOW EITHER OF THE REQUESTED VOLUME SERIAL NUMBERS. (SEE NOSBE, 4-36; SCOPE, 4-36)

FOR MULTIREEL FILES SEPARATE THE VSNI BY SLASHES.

EXAMPLES

1. SINGLE REEL FILES (DIEBOLD TAPES)

LABEL,OLDPL,L=XXXXLIBR,R,D=HY,NORING,VSN=CA0299.
LABEL,NEWPL,L=XXXXLIBR1,W,D=HY,RING,VSN=CA0298.

2. 3-REEL FILE (DIEBOLD TAPES)

VSN,MISFILE=CA0388/CA0485/CB1095.
REQUEST,MISFILE,HY,NORING. (CA0388,CA0485,CB1095)

3. SLOT TAPE WITH ACTUAL LABEL SG0402

VSN(TAPE55=SLOT14=SG0402) ** SLOTXX PRECEDES STICKER LABEL
...
LABEL,TAPE55,L=CASGOATA55,R,D=HY,NORING. (SLOT14,SG0402)

4. 2-REEL FILE OF SLOT TAPES

PAUSE. MULTI-REEL FILE WRITING SLOT02/SLOT19.
VSN,TAPE9=SLOT02=PUVZ01/SLOT19=PUVZ02.
...
LABEL,TAPE9,L=PUVZLONG,W,D=HY,RING.

*** LABELLED TAPES ***

SI TAPES SHOULD ALL BE LABELLED. THE LABEL CARD IS USED INSTEAD OF A REQUEST CARD. SEE THE TAPE LIBRARIAN, CODE 1896, (202) 227-1227, FOR ASSIGNMENT OF 6000 VRNO (VISUAL REEL NUMBER).

LABEL(LFN,L=XXXXN,R,NORING,D=Z,F=Z,T=Z,VS=VRNO)

LFN IS THE LOGICAL FILE NAME (SAME AS ON OTHER CARDS
REFERENCING THE FILE)

XXXXN IS FILE LABEL NAME UP TO 17 CHARACTERS.
FIRST FOUR SHOULD BE XXXX FROM CHARGE CARD.

R IS W WHEN WRITING THE LABEL ON TAPE USING SUPPLIED
PARAMETERS.
R WHEN READING LABEL, WILL CHECK AGAINST SUPPLIED
PARAMETERS.

NORING IS RING IF TAPE IS TO BE WRITTEN
NORING IF TAPE IS TO BE READ ONLY (DEFAULT)

D= TAPE AND LABEL DENSITY
HY 800 BPI 7-TRACK TAPE
HI 556 BPI 7-TRACK TAPE
LO 200 BPI 7-TRACK TAPE
HD 800 BPI 9-TRACK TAPE (NO L TAPES)

F= F PARAMETER OMITTED FOR SI TAPE.
S STRANGER TAPE (FOR OFFLINE DEVICE SUCH AS SC4060)
L LONG STRANGER TAPE

T= RETENTION PERIOD (1 TO 3 NUMBERS) (DEFAULT IS 1 DAY)
SYSTEM WILL NOT ALLOW 'W' (REWRITE LABEL) IF LABEL
RETENTION HAS NOT EXPIRED.

VS=VRNO 6-CHARACTER VISUAL REEL NUMBER (CABINET, SHELF, SLOT)
FOR INTERNAL CHECKING. OPTIONAL IF VS CARD IS USED.
MUST USE SEPARATE VS CARD FOR MULTIREEL OR SLOT TAPES.

COMMENTS TO THE OPERATOR MAY BE WRITTEN BETWEEN COL 50 AND COL 70.

9-TRACK TAPE USE MAY REQUIRE ADDITIONAL PARAMETERS (SEE NOSBE, 4-50;
SCOPE, 4-32). (E.G., N=EB INDICATES EBCDIC CHARACTERS)

*** UNLABELLED TAPES ***

UNLABELLED TAPES SHOULD BE AVOIDED, IF POSSIBLE. THEY ARE OFTEN USED TO TRANSFER FILES BETWEEN THE CDC 6000 AND OTHER HARDWARE. WHEN THEY ARE REQUIRED, THE REQUEST CARD IS USED INSTEAD OF A LABEL CARD.

REQUEST,LFN,DT,NORING,TYPE. (VRNO)
REQUEST,LFN,DT,NORING,TYPE,VSN=VRNO.

WHERE

DT IS THE DENSITY
HY - 800 BPI 7-TRACK TAPE (HYPER DENSITY)
HI - 556 BPI 7-TRACK TAPE (HIGH DENSITY)
HD - 800 BPI 9-TRACK TAPE (HIGH DENSITY) (NO L TAPES)

NORING IS RING IF TAPE IS TO BE WRITTEN
NORING IF TAPE IS TO BE READ ONLY (DEFAULT)

TYPE IS THE DATA FORMAT

S - A STRANGER TAPE WITH PHYSICAL RECORD SIZE NOT
EXCEEDING 512 WORDS.
L - A STRANGER TAPE WITH RECORDS OVER 512 WORDS
(7-TRACK ONLY)

VSN=VRNO 6-CHARACTER VISUAL REEL NUMBER (CABINET, SHELF,
NUMBER). FOR INTERNAL CHECKING. OPTIONAL IF VSN
CARD IS USED. MUST USE SEPARATE VSN CARD FOR
MULTIREEL OR SLOT TAPES.

FOR SI TAPES (CODED OR BINARY), USER MUST OMIT 'TYPE' KEYWORD.

AN ADDITIONAL PARAMETER, EB, IS NEEDED FOR 9-TRACK TAPES IN EBCDIC.

IF VSN=VRNO IS NOT USED, A SEPARATE VSN CARD IS REQUIRED.

*** DESCRIBING A FILE ***

EACH FILE ACCESSED BY RECORD MANAGER HAS AN ASSOCIATED FILE INFORMATION TABLE (FIT) WHICH DEFINES THE FILE AND HOW IT IS ACCESSED. THE COMPILERS DEFINE THE FIT'S REQUIRED FOR THE PROGRAM AND INITIALIZE THEM TO DEFAULT OR USER-SUPPLIED VALUES. AT EXECUTION TIME, THIS INFORMATION MAY BE CHANGED BY MEANS OF THE 'FILE' CONTROL CARD. IN THIS WAY, FILES CREATED BY A PROGRAM WRITTEN IN ONE LANGUAGE MAY BE READ BY ONE WRITTEN IN ANOTHER LANGUAGE (E.G., FILES WRITTEN BY COBOL MAY BE READ BY FORTRAN PROGRAMS).

FILE,LFN,<PARAMETERS>.

WHERE LFN IS THE FILE WHOSE FIT IS TO BE MODIFIED
<PARAMETERS> ARE THE VARIOUS FIELDS OF THE FIT WHICH MAY BE USER-DEFINED. ALL ARE IN KEYWORD FORM (KEYWORD=VALUE).

A FEW OF THE KEYWORDS ARE:

| | |
|--------|--|
| BFS | BUFFER LENGTH (TAPE FILE) |
| BT | BLOCK TYPE |
| CM | CONVERSION MODE (TAPE FILE) |
| EO | ERROR OPTION (TAPE PARITY ERRORS) |
| ERL | MAXIMUM NON-FATAL ERROR LIMIT |
| FO | FILE ORGANIZATION |
| KL | KEY LENGTH |
| LFN | REPLACEMENT FILE NAME (SEE LAST EXAMPLE BELOW) |
| MBL | MAXIMUM BLOCK LENGTH |
| MNR | MINIMUM RECORD LENGTH (VARIABLE LENGTH RECORDS) |
| MRL/FL | MAXIMUM RECORD LENGTH |
| RB | NUMBER OF RECORDS PER BLOCK |
| RT | RECORD TYPE |
| SDS | SYSTEM DAYFILE STATISTICS |

(SEE FTN, 16-6; FTN/S, III-5-6; RMC0B, APPENDIX B; RMFTN, APPENDIX C)

SAMPLE FILE CARDS

FILE,PRT,BT=C,RT=Z,MRL=150.

ZERO-BYTE TERMINATED FILE FOR PRINTING

FILE,STRANG,RB=10,MRL=80,MBL=800,RT=F,BT=K,EO=A,ERL=25,BFS=512,CM=YES.

STRANGER BLOCKED CODED TAPE

FILE,SIS,FO=IS,RT=F,KT=S,KL=10,MRL=100,RB=6,SDS=YES.

INDEXED SEQUENTIAL FILE

FILE,UNBL,BT=C,RT=S. FORTRAN BINARY 3.3 COMPATIBLE (UNBLOCKED) FILE

FILE,INPUT,LFN=DATA. SUBSTITUTE ALTERNATE INPUT FILE FOR COBOL PROG.

A LDSET CARD (CCRM, 2-16) IS NEEDED TO MAKE THE INFORMATION ON THE FILE CARD AVAILABLE TO A USER PROGRAM 'LDSET,FILES=LFN.'.

*** INDEXED SEQUENTIAL ***

CYRER RECORD MANAGER WITH FILE ORGANIZATION TYPE KNOWN AS INDEXED SEQUENTIAL (FO=IS) REPLACED FORMER SCOPE INDEXED SEQUENTIAL (SIS) VERSION 2.0.

SCOPE 3.3 COBOL SOURCE PROGRAMS USING SIS VIA 'ENTER ROUTINE-NAME' WILL COMPILE AND EXECUTE CORRECTLY UNDER SCOPE 3.4 IF THE Z OPTION FOR SCOPE 3.3 COMPATIBILITY IS SELECTED ON THE COBOL CONTROL CARD. IF THE Z OPTION IS NOT SELECTED, THE PROGRAM MUST BE CHANGED TO USE THE NEW LANGUAGE SPECIFICATIONS TO DESCRIBE AND PROCESS AN INDEXED SEQUENTIAL FILE.

RUN STATISTICS ON FILE USAGE (E.G., NUMBER OF ACCESSSES) ARE PLACED IN THE ERROR FILE ZZZZZEF. IF THE USER DESIRES THE STATISTICS IN THE DAYFILE, USE A FILE CARD TO INDICATE SDS=YES.

EXAMPLE--THE FOLLOWING ILLUSTRATES EXECUTION OF COBOL PROGRAM WHICH CREATES A SISFILE WITH THE LOCAL FILE NAME OF MYFIL AND UTILIZES SOME OF THE ABOVE DESCRIBED OPTIONS.

| | NAME / CODE |
|---------------------|-------------|
| JOBNAME,..... | |
| CHARGE,..... | |
| COBOL(LR,U,DB,Z) | |
| FILE(MYFIL,SDS=YES) | |
| LDSET(FILES=MYFIL) | |
| LGO. | |
| ' 7/8/9 | FOR |
| " 6/7/8/9 | EOF |

THE INDEXED SEQUENTIAL FILE UTILITIES DUMPS, DUMPC, CREATE, DUMPR, AND RELOADR ARE NOT INCLUDED IN NOS/BE 1.0 OR SCOPE 3.4 BECAUSE SIMILAR CAPABILITIES ARE OFFERED BY RECORD MANAGER AND FORM, AND BY NOS/BE UTILITIES. THE UTILITIES ESTMATE AND SISTAT ARE AVAILABLE.

IF A SIS FILE IS TO BE SAVED FOR BACKUP AND THE STRUCTURE OF THE FILE IS TO BE UNCHANGED WHEN RELOADED, THE COPYBF FILE UTILITY MAY BE USED TO DUMP AND RELOAD THE FILE. FILES WHICH WERE DUMPED USING DUMPR UNDER SCOPE 3.3 ARE NOT RELOADABLE UNDER NOS/BE 1.0.

THE FIRST TIME A SCOPE 3.3 IS FILE IS OPENED BY A NOS/BE 1.0 PROGRAM, THE FILE AUTOMATICALLY UNDERGOES A CONVERSION (THE FSTT IS CHANGED) THAT MAKES IT NO LONGER PROCESSABLE BY SCOPE 3.3 BINARIES. THUS, IF THE FILE IS USED BY A SERIES OF PROGRAMS AND A CHANGE IS MADE TO ONE OR MORE SOURCE PROGRAMS, ALL OF THE PROGRAMS MUST BE RECOMPILED.

THE FOLLOWING EXAMPLES ILLUSTRATE THE USE OF FORM TO PERFORM SCOPE 3.3 SIS UTILITY FUNCTIONS. (SEE FORM, 4-4, 5-14)

IN THIS EXAMPLE THE SEQUENTIAL RECORDS OF A SIS FILE ARE COPIED TO A SEQUENTIAL FILE.

```

JOBNAME,.....          NAME / CODE
CHARGE,.....
FILE(SISFILE,FO=IS,KL=10,KT=I,MNR=1000,MRL=1000)
FILE(SEQFILE,FO=SQ,BFS=260,LT=UL,BT=C,RT=Z,MRL=1000)
LDSET(FILES=SISFILE/SEQFILE)
FORM.
' 7/8/9      FOR
INP(SISFILE)
OUT(SEQFILE)
XFQ(FIN)
" 6/7/8/9    EOF

```

IN THIS EXAMPLE A SIS FILE IS CREATED FROM A SEQUENTIAL INPUT FILE. THE OUT DIRECTIVE IN THE FORM CONTROL CARDS INDICATES THE KEY TO BE USED IN BUILDING THE SIS FILE. IN THIS CASE IT WILL BE A 10-CHARACTER INTEGER STARTING WITH CHARACTER 1 OF THE INPUT RECORD.

```

JOBNAME,.....          NAME / CODE
CHARGE,.....
FILE(SEQFILE,BFS=260,LT=UL,BT=C,RT=Z,MRL=1000,FO=SQ)
FILE(SEQFILE,KT=I,KL=10)
FILE(SISFILE,FO=IS,KL=10,MNR=1000,MRL=1000,KT=I,ERL=10)
FILE(SISFILE,DP=20,IP=20)
LDSET(FILES=SISFILE/SEQFILE)
FORM.
' 7/8/9      EOR
INP(SEQFILE)
OUT(SISFILE,KEY=+1I10)
XFQ(FIN)
" 6/7/8/9    EOF
FILE CARD AVAILABLE TO A USER PROGRAM 'LDSET,FILES=LFN.'

```

***** FORTRAN EXTENDED *****

*** INTRODUCTION ***

NOS/BE AT DTNSRDC HAS TWO FORTRAN COMPILERS: FORTRAN IV EXTENDED (FTN) AND THE 'RUN' COMPILER (WHICH IS NO LONGER SUPPORTED BY CDC OR DTNSRDC). RUN FORTRAN IS NOT DISCUSSED FURTHER IN THIS CHAPTER. FOR DETAILS OF THE FTN LANGUAGE, SEE PAGE I: FTN OR FTN/S. ALL THE SPECIAL LIBRARY ROUTINES (SIN, COS, ...) ARE FOR FTN ONLY. THERE ARE SOME ALTERNATIVE FTN-COMPATIBLE COMPILERS DISCUSSED AT THE END OF THIS CHAPTER.

*** FTN CONTROL CARD PARAMETERS ***
 (SEE FTN, CHAPTER 10 OR FTN/S, CHAPTER I-11)

EXAMPLES FTN. DEFAULTS TO (A,B=LGO,BL=0,EL=I,ER,I=INPUT,L=OUTPUT,
 OPT=0,P=0,PD=6,PL=5000,R=1,ROUND=*/,SL,T)
 FTN(GO) COMPILE AND EXECUTE
 FTN,I=MYFILE,ROUND,OPT=1.
 FTN(B=PUNCHB,EL=A)

| OPTION | ACTION |
|----------|--|
| A | ABORT JOB IF ERRORS OCCUR DURING COMPILATION. NEXT CONTROL CARD TO BE EXECUTED IS THE ONE FOLLOWING 'EXIT,S.' IF NO 'EXIT,S.' CARD, JOB ENDS. (DEFAULT) |
| A=0 | CONTINUE EVEN IF ERRORS IN COMPILATION (FORCED IF D USED) |
| B=PUNCHB | PRODUCE PUNCHED BINARY DECKS OF ALL ROUTINES; NO BCD SEQUENCING AT END OF CARDS |
| B=LFN | PUT BINARY IMAGES ON FILE 'LFN'; IF OMITTED, LFN=LGO |
| B=0 | NO BINARY OUTPUT |
| BL | BURSTABLE LIST. EACH MAJOR SECTION OF COMPILATION LISTING STARTS ON A NEW PAGE. |
| BL=0 | COMPACT LIST. NEW PAGE FOR FIRST PAGE ONLY. (DEFAULT) |
| D | DEBUG MODE (FORCES A=0, OPT=0, T) SPECIAL COMMENT CARDS (C IN COLUMN 1 AND \$ IN COLUMN 2) MAY BE USED TO AID IN DEBUGGING FTN PROGRAMS WHEN THE 'D' OPTION IS SPECIFIED. THESE FEATURES INCLUDE ARRAY BOUND CHECKING, DUMPING VARIABLES WHEN CHANGED BY A REPLACEMENT STATEMENT, AND INTER- AND INTRA-PROGRAM TRACING. (SEE FTN, CHAPTER 9; FTN/S, CHAPTER I-13; DEBUG) |
| EL=F | LIST FATAL DIAGNOSTICS |
| =I | SAME AS F PLUS INFORMATIVE DIAGNOSTICS (DEFAULT) |
| =A | SAME AS I PLUS DIAGNOSTICS INDICATING NON-ANSI USAGE |
| ER | GENERATE CODE FOR OBJECT-TIME REPRIEVE OF ERRORS (DEFAULT FOR TS AND OPT=0) |
| ER=0 | DO NOT GENERATE REPRIEVE CODE (DEFAULT FOR OPT=1,2) |
| GO | LOAD AND EXECUTE OBJECT CODE WITHOUT A SEPARATE 'LGO.'. |
| I=LFN | FORTRAN SOURCE INPUT IS ON 'LFN' INSTEAD OF 'INPUT' |
| I | SAME AS I=COMPILE (SEE CCRM, 7-2, 7-3) |

L=LFN OUTPUT LISTS (BL, EL, OL, R, SL) ARE TO BE PUT ON FILE
 'LFN' RATHER THAN 'OUTPUT'.
L=0 ALL LISTINGS ARE SUPPRESSED, EXCEPT FATAL DIAGNOSTICS AND
 THE STATEMENTS CAUSING THEM.

OL OBJECT CODE LISTING (USE ONLY WHEN REQUESTED BY 1892)

OPT=0 FAST COMPILE (DEFAULT)
 =1 STANDARD OPTIMIZATION (USE FOR PRODUCTION)
 =2 FAST OBJECT CODE (SLOWER COMPILE) (NOT RECOMMENDED)
 (ALSO SEE PARAMETERS 'Q' AND 'TS')

P PAGE NUMBERING OF OUTPUT LIST IS CONTINUOUS FROM
 SUBPROGRAM TO SUBPROGRAM.
P=0 EACH SUBPROGRAM STARTS WITH PAGE 1 (DEFAULT).

PD=8 COMPILER TIME LISTINGS ARE 8 LINES PER INCH
PD=6 COMPILER TIME LISTINGS ARE 6 LINES PER INCH (DEFAULT)
PD IMPLIES PD=8 (PRINT DENSITY)

PL=N SPECIFY DECIMAL MAXIMUM NUMBER OF UNIT RECORDS TO BE
 WRITTEN AT EXECUTION TIME ON THE FILE 'OUTPUT' (DEFAULT:
 PL=5000). THE PRINT LINE LIMIT MAY BE RESET AT EXECUTION
 TIME BY SPECIFYING PL=NNNNN, WHERE NNNNN IS THE NEW LINE
 LIMIT, ANYWHERE ON THE LGO CARD. IT DOES NOT AFFECT
 NORMAL FILE NAME REPLACEMENT. (SEE CCRM, 2-15)

PW=N PAGE WIDTH (NUMBER OF CHARACTERS PER LINE IN COMPILATION
 LISTING. $50 \leq N \leq 136$) VALID ONLY WITH TS MODE.
PW IMPLIES PW=72.

Q FAST COMPILE. NO OBJECT CODE. NO ADDRESSES IN REFERENCE
 MAP. NO EXECUTION POSSIBLE.

R=0 NO REFERENCE MAP
 =1 SHORT MAP (DEFAULT) (SYMBOLS, ADDRESSES, TYPE, DO LOOPS)
 =2 LONG MAP (SHORT MAP PLUS LINE NUMBER REFERENCES)
 =3 LONG MAP PLUS COMMON BLOCK MEMBERS AND EQUIVALENCE CLASSES

ROUND=OPS ROUNDED ARITHMETIC FOR SPECIFIED OPERATORS (ANY OR ALL OF
 +, -, *, /)
ROUND=0 ROUNDED ARITHMETIC WILL NOT BE USED
ROUND=* / DEFAULT
ROUND IMPLIES ROUND=+-*/

SL=0 DO NOT LIST SOURCE PROGRAM

T TURN ON ERROR TRACEBACK IN LIBRARY ROUTINES. (DEFAULT)
 (FORCED WHEN D OR OPT=0)
T=0 TURN OFF ERROR TRACEBACK.
 (F.G., SQUARE ROOT OF NEGATIVE NUMBER WILL BE FLAGGED AS
 AN ERROR, BUT WILL NOT SHOW WHERE THE ERROR OCCURRED.)

TS TIME-SHARING MODE. 'OPT' IGNORED IF 'TS' SPECIFIED.

ALL PARAMETERS ARE OPTIONAL. 'FTN.' WILL USE ALL DEFAULT PARAMETERS.

*** PROGRAM CARD ***

THE FIRST STATEMENT OF A FORTRAN MAIN PROGRAM MUST BE A PROGRAM CARD IN THE FOLLOWING FORMAT (FTN, 7-2; FTN/S, I-7-1):

PROGRAM NAME (F1,F2,...,FN)

WHERE 'NAME' IS THE NAME OF THE MAIN PROGRAM (FROM 1 TO 7 ALPHANUMERIC CHARACTERS, BEGINNING WITH A LETTER). THE PARAMETERS 'FJ' ARE THE NAMES OF ALL INPUT/OUTPUT FILES REQUIRED BY THE MAIN PROGRAM AND ALL SUBPROGRAMS. EACH IS 1 TO 6 ALPHANUMERIC CHARACTERS, BEGINNING WITH A LETTER. NO MORE THAN 50 FILES MAY BE USED.

IN THE DISCUSSION BELOW, 'F' IS A FORMAT NUMBER.

1. IF THE PROGRAM HAS CARD INPUT, PUNCHED CARD OUTPUT AND/OR LINE PRINTER OUTPUT, FILE NAMES 'INPUT', 'PUNCH' AND/OR 'OUTPUT', RESPECTIVELY, MUST APPEAR IN THE LIST.
2. THE FILE NAME 'TAPEJ' MUST APPEAR IF READ(J,F), WRITE(J,F), READ(J), WRITE(J), BUFFER IN(J,...), OR BUFFER OUT(J,...) STATEMENTS ARE USED. 'J' IS AN INTEGER FROM 1-99. IF 'J' IS A VARIABLE, THERE MUST BE A FILE NAME 'TAPEJ' FOR EACH VALUE 'J' MAY ASSUME.
3. ANY FILE NAME MAY BE EQUATED TO ANY OTHER FILE NAME WHICH PRECEDES IT IN THE LIST. FOR EXAMPLE, IF UNIT 5 IS TO BE THE CARD READER (READ(5,F)<LIST>), THEN 'TAPE5=INPUT' MUST APPEAR IN THE LIST AFTER THE DEFINITION OF 'INPUT'. SIMILARLY, IF UNIT 6 IS THE LINE PRINTER (WRITE(6,F)<LIST>), THEN 'TAPE6=OUTPUT' MUST APPEAR IN THE LIST AFTER THE DEFINITION OF 'OUTPUT'.
4. THE FILE 'DEBUG=OUTPUT' SHOULD APPEAR ON THE PROGRAM CARD IF THE FORTRAN DEBUG OPTION MIGHT BE INVOKED AND THE USER DESIRES INTERSPERSED DEBUG OUTPUT. OTHERWISE THE DEBUG FILE WILL BE DISPOSED TO THE PRINTER BY THE SYSTEM AT END-OF-JOB.
5. IF A FILE WILL ALWAYS RESIDE ON DISK (NEVER ON TAPE), THE STANDARD BUFFER SIZE OF 512 (DECIMAL) MAY BE DECREASED. FORMATTED FILES MAY BE REDUCED TO 128 (E.G., 'TAPE15=128'); CONNECTED INTERCOM FILES TO 64; BUFFERED FILES TO 0. THIS IS ESPECIALLY USEFUL IN INTERCOM PROGRAMS WHERE SPACE MAY BE CRITICAL.
6. IF A FILE MAY RESIDE ON TAPE, INCREASING THE BUFFER SIZE TO 1024 (DOUBLE BUFFERING) SHOULD IMPROVE RUNNING TIME.
7. THE DEFAULT MAXIMUM LENGTH FOR FORMATTED, LIST-DIRECTED AND NAMELIST RECORDS IS 150. THIS MAY BE CHANGED BY INCLUDING A NEW MAXIMUM LENGTH AFTER THE FILE NAME (E.G., 'TAPE1=/200', OR 'TAPE26=128/50').
8. AT EXECUTION TIME, FILE NAMES MAY BE REPLACED IN THE ORDER IN WHICH THEY APPEAR IN THE PROGRAM CARD. EQUATED FILES SHOULD APPEAR AFTER OTHER FILE DEFINITIONS, SINCE THEY MAY NOT BE REPLACED. (CCRM, 10-13)

EXAMPLES - PROGRAM CAT (INPUT,OUTPUT,TAPE18,TAPE5=INPUT,TAPE6=OUTPUT)
PROGRAM TERM (INPUT=128,OUTPUT=128,TAPE13,TAPE5=INPUT)
PROGRAM CHECK (OUTPUT=128,TAPE1=128/200,TAPE2=/50)

*** CONVERSION TO FORTRAN EXTENDED ***

1. ADD THE 'PROGRAM' STATEMENT WITH ALL FILE NAMES LISTED.
 2. COMPILE THE PROGRAM WITH DEFAULT 'OPT=0' UNTIL IT IS ERROR-FREE. PRODUCTION PROGRAMS SHOULD BE COMPILED WITH 'OPT=1'.
 3. DIVISION BY ZERO DOES NOT YIELD A ZERO AS SOME OTHER COMPUTERS DO, BUT USING THE RESULT WILL CAUSE A HALT IN EXECUTION. THE BUILT-IN FUNCTION 'LEGVAR(V)' WILL TEST FOR INFINITE AND INDEFINITE QUANTITIES. THE FUNCTION WILL RETURN -1 FOR AN INDEFINITE, +1 FOR AN INFINITE, AND 0 FOR NORMAL. USER MAY ALSO TEST FOR THE DIVISOR EQUAL TO ZERO AND, IF SO, SET THE RESULT TO ZERO, IF DESIRED. (FTN, 8-15; FTN/S, I-8-15)
 4. ALTER ANY USE OF THE FOLLOWING MACHINE-DEPENDENT FEATURES:

| IBM 7090 | IBM S/360 | CDC 6000 |
|-------------------|------------------------|-------------------|
| ----- | ----- | ----- |
| 36 BITS/WORD | 32 BITS/WORD | 60 BITS/WORD |
| 12 OCTAL DIG/WORD | 8 HEXADECIMAL DIG/WORD | 20 OCTAL DIG/WORD |
| 6 CHAR/WORD | 4 CHAR/WORD | 10 CHAR/WORD |
- FORMATS OF A8 OR LESS NEED NOT BE CHANGED IF THEY CONSTITUTE ALPHANUMERIC INFORMATION FOR PURPOSES OF PRINTING ONLY.
5. SPECIFICATION STATEMENTS (DIMENSION, EQUIVALENCE, COMMON AND TYPE) MUST APPEAR BEFORE DATA, ARITHMETIC STATEMENT FUNCTION DEFINITION AND NAMELIST STATEMENTS, WHICH IN TURN MUST COME BEFORE ANY EXECUTABLE STATEMENT.
 6. DOUBLE PRECISION IBM S/360 PROGRAMS MAY BE CONVERTED TO SINGLE PRECISION FOR CDC 6000 SERIES WITH A SNOBOL PROGRAM IN FILE CVT360 (SEE CCRM, 10-21). EBCDIC CARDS MAY BE USED DIRECTLY (SEE CCRM, 1-6).

*** PITFALLS IN FORTRAN ***

NUMBERS IN PARENTHESES REFER TO THE PAGE IN FTN MANUALS (NOS/BE MANUAL FIRST / SCOPE MANUAL SECOND).

1. OCTAL CONSTANTS USED IN DATA STATEMENTS SHOULD NOT BE PRECEDED BY THE LETTER O BUT SHOULD BE FOLLOWED BY THE LETTER B. (1-8/I-2-5)
THE LETTER O IS A VALID FORMAT SPECIFICATION (E.G., O20 MAY BE USED TO WRITE OUT A 60-BIT WORD IN OCTAL).
2. DOUBLE PRECISION CONSTANTS ARE DEFINED BY THE D EXPONENT. DUE TO THE 48-BIT COEFFICIENT IN CDC 6000 SINGLE PRECISION, FEW OPERATIONS REQUIRE DOUBLE PRECISION.
3. SOME INTEGER MULTIPLICATION AND DIVISION IS DONE IN FLOATING POINT MODE. HENCE, SHIFTING DATA IN FORTRAN BY MULTIPLYING OR DIVIDING BY POWERS OF TWO WILL NOT ALWAYS WORK. A FORTRAN BUILT-IN FUNCTION 'SHIFT(A,I)' WILL PERFORM THIS OPERATION. THE FUNCTION WILL SHIFT 'A' 'I' BITS LEFT CIRCULAR IF 'I' IS POSITIVE; 'I' BITS RIGHT WITH SIGN EXTENSION AND END-OFF IF 'I' IS NEGATIVE.
4. MIXED-MODE EXPRESSIONS (E.G., A=B+2) ARE ALLOWED AND NO ERROR DIAGNOSTIC WILL RESULT. HOWEVER, THIS IS NOT STANDARD FORTRAN AND SHOULD BE AVOIDED.
5. HOLLERITH STRINGS IN FORMAT STATEMENTS MAY BE DELIMITED BY DOUBLE QUOTES (4-8 PUNCH 026; 7-8 PUNCH 029) OR ASTERISKS OR DEFINED BY HOLLERITH COUNT (USING H). IN DATA OR ARGUMENT LISTS, ONLY HOLLERITH COUNT OR DOUBLE QUOTES MAY BE USED.
6. ON THE CDC 6000, THERE IS NO WAY TO EXPLICITLY DEFINE THE LENGTH OF A VARIABLE. (E.G., 'REAL*8 MASS' MUST BE CHANGED TO 'REAL MASS' OR 'DOUBLE PRECISION MASS'.)
7. A VARIABLE NAME WITHIN A SUBROUTINE CANNOT BE THE SAME AS THE SUBROUTINE NAME. (7-7/I-7-13)
8. DUMMY ARGUMENTS OF A SUBROUTINE MAY NOT APPEAR IN A COMMON, DATA OR EQUIVALENCE STATEMENT IN THE SUBROUTINE. (7-12/I-7-12)
9. ALL ARGUMENTS TO SUBROUTINE OR FUNCTION SUBPROGRAMS ARE TRANSFERRED BY LOCATION (NAME) ON THE CDC 6000 (EXCEPT FOR LIBRARY ROUTINES, WHICH WILL HAVE VALUES TRANSMITTED BY VALUE UNLESS T, D OR OPT=0 IS INVOKED BY THE FTN CONTROL CARD OR THE EXTERNAL FUNCTION NAME APPEARS IN AN EXTERNAL STATEMENT IN THE CALLING PROGRAM). SLASHES USED IN IBM S/360 SUBROUTINES TO PASS UNDIMENSIONED ARGUMENTS BY NAME MUST BE REMOVED.

10. FTN 4 ENTRY POINTS ARE HANDLED DIFFERENTLY FROM OTHER FORTRAN IMPLEMENTATIONS. THE ARGUMENT LIST, IF ANY, APPEARING WITH THE FUNCTION OR SUBROUTINE STATEMENT DOES NOT APPEAR WITH THE ENTRY STATEMENT BUT IS ASSUMED TO BE THE SAME. IN THE CALLING PROGRAM, THE REFERENCE TO THE ENTRY NAME INCLUDES THE ARGUMENT LIST. IN A MULTIPLE ENTRY FUNCTION, REFERENCE IS ALWAYS TO THE FUNCTION NAME, RATHER THAN THE ENTRY NAME (7-18/I-7-22). NOTE THAT ALL SPECIFICATION STATEMENTS MUST PRECEDE ALL EXECUTABLE STATEMENTS, HENCE NO SPECIFICATIONS MAY FOLLOW AN ENTRY POINT. FOR EXAMPLE, CALLS IN THE MAIN ROUTINE MIGHT BE AS FOLLOWS:

```

.....
L = JOE(3.*P,Q)
.....
K = JAM(R,A)

```

WHERE THE SUBPROGRAM IS /S FOLLOWS
FUNCTION JOE(X,Y)

```

.....
JOE = Y-X
RETURN
ENTRY JAM
.....
JOE = X-Y
RETURN
END

```

NOTE THAT FTN VERSION 5 (THE ANSI PROPOSED STANDARD) WILL IMPLEMENT THIS DIFFERENTLY.

11. FTN 4 MULTIPLE RETURNS FROM SUBPROGRAMS ARE IMPLEMENTED DIFFERENTLY FROM OTHER FORTRAN COMPILERS AS THE FOLLOWING EXAMPLE SHOWS.
(7-6/I-7-14)

```

.....
CALL PGM1 (A,B,C), RETURNS (5,6)
.....
5 D=A*C
.....
6 DO 10 I=1,25
.....

```

WHERE THE SUBROUTINE IS
SUBROUTINE PGM1 (X,Y,Z), RETURNS (M,N)

```

.....
RETURN M
.....
RETURN N
.....
RETURN
END

```

NOTE THAT FTN VERSION 5 (THE ANSI PROPOSED STANDARD) WILL IMPLEMENT THIS DIFFERENTLY.

12. MISSING SUBPROGRAMS DO NOT INHIBIT EXECUTION OF A SIMPLE OR OVERLAY PROGRAM. IF A MISSING ROUTINE IS ACTUALLY CALLED, THE RUN WILL ABORT WITH A DAYFILE MESSAGE 'ERROR MODE 1, LOCATION 4XXXXX' (XXXXXX IS THE ADDRESS WHERE THE MISSING ROUTINE WAS CALLED. UNSATISFIED REFERENCES WILL BE LISTED IN THE STORAGE MAP.

13. SOME OF THE IBM S/360 AND IBM 7090 BUILT-IN FUNCTIONS ARE NOT NAMED THE SAME OR ARE NON-EXISTENT ON THE CDC 6000 LIBRARY. THOSE MISSING FROM THE LIBRARY MAY BE OBTAINED FROM CODE 1892. (CHAP 8/CHAP I-8)

| 7090 | 360 | 6000 | USE |
|--------|--------|--------|--------------------------|
| ASIN | ARSIN | ASIN | |
| ACOS | ARCOS | ACOS | |
| COTAN | COTAN | | |
| ERF | ERF | ERF | |
| GAMMA | GAMMA | GAMMA | |
| ALGAMA | ALGAMA | | |
| TIMER | | SECOND | (CPU TIME) |
| TIMER | | CLOCK | (SAME AS SECOND) |
| | | TIME | (CLOCK TIME - HH.MM.SS.) |
| LOC | | LOCF | (ADDRESS OF ARGUMENT) |
| | | RANF | (RANDOM NUMBER) |

14. ARRAYS MAY HAVE ONE, TWO OR THREE SUBSCRIPTS. VARIABLES IN SUBSCRIPTS OF THREE DIMENSIONS CAUSE SLOWER EXECUTION, ESPECIALLY IN DOUBLE PRECISION OR COMPLEX ARRAYS.

15. OVERFLOW AND DIVIDE CHECK TESTS ARE NON-EXISTENT ON THE CDC 6000. USE THE 'LEGVAR' FUNCTION TO TEST FOR THESE SITUATIONS. (8-14/I-8-15)

16. WHEN A COMPUTED GO TO COUNTER IS OUTSIDE THE RANGE, 6000 FORTRAN EXTENDED WILL ABORT THE RUN. THE COUNTER MAY BE ANY EXPRESSION BUT WILL BE TRUNCATED TO AN INTEGER AFTER EVALUATION AND BEFORE THE TEST.

17. DO LOOPS CANNOT TERMINATE WITH AN 'IF' STATEMENT. (4-8/I-5-8)

18. TWO-BRANCH ARITHMETIC AND LOGICAL 'IF' STATEMENTS ARE AVAILABLE IN CDC FORTRAN EXTENDED. (4-6, 8/I-5-6, 8)

19. ENCOUNTERING AN END-OF-FILE WILL NOT TERMINATE A JOB UNTIL AN ATTEMPT IS MADE TO READ BEYOND THE EOF. THE FIRST VARIABLE TO BE READ IN THE CURRENT UNIT RECORD WHEN AN EOF HAS BEEN ENCOUNTERED WILL BE SET TO -ZERO, OTHERS ARE UNCHANGED. THE FUNCTION 'EOF(N)' MAY BE USED IN AN 'IF' TEST TO SENSE AN EOF ON TAPEN. AFTER TESTING, READING THE NEXT FILE IS ALLOWED. THE FUNCTION 'IOCHECK(I)' MAY BE USED TO CHECK FOR READING REDUNDANCIES. THESE FUNCTIONS REPLACE THE 'END=' AND 'ERR=' PARAMETERS IN THE IBM S/360 READ STATEMENT, WHICH ARE NOT AVAILABLE. (8-23/I-8-19) 'END=' AND 'ERR=' ARE A PART OF THE PROPOSED ANSI FORTRAN 77 STANDARD.

20. ATTEMPT TO PRINT NUMBERS INVALID FOR THE FORMAT FIELD WILL CAUSE ONE OF THE FOLLOWING RESULTS:

- * IN THE ENTIRE FIELD IF THE NUMBER IS TOO LARGE FOR THE FIELD.
- R IF VALUE IS INFINITE (RANGE ERROR)
- I IF VALUE IS INDEFINITE

(6-8, 9/I-10-9, 13)

21. THE FORMAT SPECIFICATION 'T' FOR TABULAR COLUMN SPECIFICATION IS AVAILABLE ON THE CDC 6000.

22. NAMELIST DATA ON THE CDC 6000 IS THE SAME AS ON THE IBM 7090. IBM S/360 NAMELIST DATA MUST BE CHANGED TO ENCLOSE EACH STRING IN DOLLAR SIGNS (E.G., \$NAM1...\$).

23. REFORMATTING IN CORE IS POSSIBLE BY MEANS OF 'ENCODE' AND 'DECODE' STATEMENTS. THIS METHOD IS SLOW. (5-12/I-9-12)

24. FORMATTED OR NAMELIST INPUT AND OUTPUT HAS A DEFAULT LIMIT OF 150 CHARACTERS PER UNIT RECORD. PRINTER OUTPUT GREATER THAN 137 CHARACTERS IS PRINTED ON 2 LINES. THE SIZE MAY BE ALTERED IN THE PROGRAM CARD (E.G., 'TAPE1=/400,TAPE2=/121') (7-2/I-7-2); OR DIRECT CALLS TO RECORD MANAGER MAY BE USED (8-39/III-6); OR 'BUFFER IN', 'BUFFER OUT', 'DECODE' AND 'ENCODE' STATEMENTS MAY BE USED, ALONG WITH 'UNIT' AND 'LENGTH' FUNCTIONS (8-23/I-9-13,14, I-8-18,19), TO PERFORM FORMATTED I/O FOR RECORDS IN EXCESS OF THE LIMITATIONS SPECIFIED ABOVE.

25. RANDOM ACCESS TO DATA ON MASS STORAGE IS ACCOMPLISHED BY CALLABLE SYSTEM ROUTINES (E.G., READMS). (8-28/I-8-14, III-7-1) SINCE SUCH RANDOM DATA IS NOT BUFFERED AHEAD, IT IS SLOW. 'OPENMS' IS CALLED TO OPEN THE RANDOM FILE AND SET UP THE INDEX POINTERS. THE USER MUST NOT ZERO THE INDEX ARRAY AFTER OPENMS. 'CLOSMS' MAY BE CALLED TO CLOSE THE FILE, BUT IT IS NOT REQUIRED. BLOCK AS WELL AS INDEX LENGTH SHOULD BE A POWER OF 2 FOR SPEED.

26. ENCODE/DECODE BUFFER SIZE IS RESTRICTED TO 150 CHARACTERS OR LESS. MULTIPLE ENCODE/DECODE STATEMENTS ARE REQUIRED FOR BUFFERS GREATER THAN 150 CHARACTERS.

27. THE MAIN CARRIAGE CONTROL CHARACTERS AND THEIR ACTIONS ARE:

| CHARACTER | ACTION BEFORE PRINTING | |
|-----------|---------------------------|--|
| ----- | ----- | |
| 1 | EJECT TO TOP OF NEW PAGE | |
| (BLANK) | SPACE 1 | |
| + | NO SPACE | |
| 0 (ZERO) | SPACE 2 | (1 BLANK LINE) |
| -(MINUS) | SPACE 3 | (2 BLANK LINES) (NOT ON 200UT TERMINALS) |
| 2 | SKIP TO LAST LINE ON PAGE | (NOT ON 200UT TERMINALS) |

WHEN Q, R, S OR T IS USED AS CARRIAGE CONTROL, NO PRINTING TAKES PLACE. THE REMAINDER OF THE LINE WILL BE IGNORED. (6-31/I-10-32; CCRM, 14-4; NOSBE, 3-40; SCOPE, APPENDIX F-2)

'PM' MAY BE USED IN COLUMNS 1-2 OF AN OUTPUT PRINT LINE FOR DISPOSITION OF REMOTE FILES. (SEE CCRM, 14-3)

28. CURRENT CORE BUFFERS FOR WRITTEN FILES NOT EQUATED TO 'OUTPUT' ARE NOT FLUSHED (EMPTIED) TO THE FILE IF A JOB ABORTS. THUS, LINES OF DATA MAY BE LOST. 'RECOVR' (8-12/I-8-13; CCRM, 4-16; NOSBE, 6-22; SCOPE, 2-20) MAY BE USED.

29. IF MORE THAN ONE BLOCK DATA SUBPROGRAM IS USED, EACH MUST HAVE A NAME (7-5/I-6-25).

30. WHEN COMPASS ROUTINES ARE INTERMIXED IN FORTRAN CODE, THE MINIMUM FIELD LENGTH REQUIRED IS 55000 TO ALLOW FOR USE OF COMPASS PSEUDO-OPS.

31. IF THE PRIMARY ENTRY NAME OF A MULTIPLE ENTRY SYSTEM LIBRARY ROUTINE IS REPLACED BY A ROUTINE OF THE SAME NAME IN USER'S PROGRAM (SUCH AS XAXISV IN THE SC4020 PACKAGE), ALL SECONDARY ENTRIES (IF REFERENCED) MUST BE REPLACED ALSO. UNSATISFIED EXTERNALS WILL RESULT OTHERWISE.

32. IF OBJECT FILE (LGO) HAS BEEN CATALOGED, EVEN THOUGH SOME FORTRAN SUBPROGRAMS DID NOT COMPILE CORRECTLY, 'COPYL' (CCRM, 6-2) MAY BE USED TO REPLACE THE BAD COMPILATIONS FROM A NEW OBJECT SUBFILE. IF THE BAD ROUTINE WAS A MAIN PROGRAM IN AN OVERLAY JOB, THE OVERLAY CARD MUST BE INSERTED BY 'COPYN' (CCRM, 6-3). WHEN AN OVERLAY CARD IS BEING COPIED, BEGIN IN COLUMN 1 AS ANY OTHER NOS/BE CONTROL CARD.

33. COMMENT CARDS SHOULD NOT OCCUR FOLLOWING THE LAST END CARD. THE COMPILER WILL PRINT THE NON-FATAL DAYFILE MESSAGE 'NULL PROGRAM IGNORED AFTER X...X', WHERE X...X IS THE NAME OF THE LAST ROUTINE.

*** FORTRAN IV EXTENDED - VERSION 4 ***

SOME DIFFERENCES BETWEEN EARLIER IMPLEMENTATIONS OF FORTRAN AND COC FORTRAN IV EXTENDED, VERSION 4, ARE:

1. FTN COMPILED PROGRAMS HAVE LINE LIMIT ON FILE 'OUTPUT' AT EXECUTION TIME. ON THE FTN OR LGO CARD, ADD PARAMETER 'PL=NNNNN' TO SET DECIMAL LINE LIMIT UP OR DOWN FROM THE DEFAULT 5000 LINES. (FTN, 10-6; FTN/S, I-11-6)

2. IMPLICIT STATEMENT ALLOWS ALL VARIABLES BEGINNING WITH STATED LETTER TO BE GIVEN THE SAME TYPE. FOR EXAMPLE, 'IMPLICIT INTEGER (F)' WOULD MAKE ALL 'F' VARIABLES (SUCH AS F, FLO, FAR) BE INTEGERS. (FTN, 3-3; FTN/S, I-6-3)

3. 'WRITMS' HAS TWO ADDITIONAL OPTIONAL PARAMETERS. THE DTNSRDC SCOPE 3.3 DEFAULT OF REWRITE IN PLACE WHERE SPACE IS AVAILABLE IS NOT THE NOS/BE DEFAULT. TO USE THIS MORE EFFICIENT REWRITE, THE USER MUST 'CALL WRITMS (U,F,N,K,-1)'. IF THE -1 PARAMETER IS OMITTED, ALL REWRITES WILL GO AT THE END OF THE FILE. (FTN, 8-30; FTN/S, III-7-2)

4. LIST-DIRECTED I/O IS AN ADDED FEATURE OF SCOPE 3.4. IT REPLACES THE TERMIN ROUTINE AND IS MORE FLEXIBLE. FOR THE FREE-FORMAT I/O, AN ASTERISK REPLACES THE FORMAT NUMBER IN READ, WRITE OR PRINT. INPUT DATA CONSISTS OF COMMA SEPARATED VALUES. ALPHANUMERIC FIELDS MUST BE ENCLOSED IN DOUBLE-QUOTE (4-8) PUNCHES. (FTN, 5-8; FTN/S, I-9-7)

5. HOLLERITH STRINGS AND EXPRESSIONS MAY OCCUR IN OUTPUT LISTS. (FTN, 6-26; FTN/S, I-9-11, I-10-1)

6. 'MOVLEV' WILL MOVE WHOLE ARRAYS FROM ONE AREA TO ANOTHER WITHOUT CONVERSION (E.G., 'CALL MOVLEV (HERE,THERE,100)'). NOTE: BECAUSE DATA IS MOVED 2 WORDS AT A TIME, 'HERE' AND 'THERE' MUST BE PHYSICALLY SEPARATED BY AT LEAST ONE WORD OF CORE (I.E., WHEN 'HERE' IS A(1), 'THERE' MAY NOT BE A(2)). IF THE NUMBER OF WORDS TO MOVE IS ODD, THE LAST MOVE IS ONLY ONE WORD. (FTN, 8-25; FTN/S, I-8-11)

7. WHEN 'BUFFER IN' IS USED TO READ STRANGER TAPES, FUNCTION 'LENGTH' MAY BE REPLACED BY A THREE-ARGUMENT SUBROUTINE 'LENGTHX', ALLOWING THE USER TO CHECK FOR OTHER COMPUTER SYSTEM WORD SIZES. (FTN, 8-24; FTN/S, I-8-19)

8. THERE IS LESS ADVANTAGE IN HAVING BLANK COMMON IN PROGRAMS, SINCE THE LOADER IS IN LOW CORE AND CANNOT BE OVERLAID BY ANY COMMON BLOCKS. SEGMENTED JOBS CANNOT BOUNCE CORE WHEN BLANK COMMON IS USED. IF ABSOLUTE PROGRAMS ARE KEPT IN A LIBRARY, BLANK COMMON CAN BE USED TO REDUCE THE PHYSICAL SIZE (NUMBER OF PRUS) OF THE ABSOLUTE MODULE.

9. FORTRAN CALLABLE 'SKPFIL' MAY BE USED TO SKIP ON TAPE (EITHER SYSTEM STANDARD OR STRANGER) OR DISK. DIFFERENT ARGUMENT LISTS ARE NEEDED TO SKIP FORWARD OR BACKWARD, LOGICAL RECORDS OR FILES. (CCRM, 4-16)

10. SEGMENTATION USING SEGMENT CARDS IN FTN DOES NOT EXIST IN NOS/BE. 'SEGLOAD', IMPLEMENTED IN SCOPE 3.4, CLOSELY RESEMBLES OVERLAY ON THE IBM S/360. (CCRM, 2-19)

11. FTN BINARY BLOCKED FILES CREATED UNDER SCOPE 3.3 MUST BE CONVERTED USING THE 'BBT06RM' UTILITY (SEE CONV). THE DEFAULT RECORD FOR BBT06RM IS NOT THE DEFAULT FOR FTN 4.6, HOWEVER. 'BBT06RM,INFILE,OUTFIL.' CREATES C-TYPE BLOCKS. TO USE THE RESULTING 'OUTFIL', INSERT BEFORE THE 'LGO' AT EXECUTION TIME:

```
FILE(OUTFIL,BT=C,RT=W)
LDSET(FILES=OUTFIL)
```

'BBT06RM(INFILE,OUTFIL,1,1)' WILL CREATE FTN DEFAULT I-TYPE BLOCKS. FOR THESE, AT LEAST 512 WORDS MUST BE IN THE FILE BUFFER. NO 'FILE' OR 'LDSET' CARDS ARE REQUIRED.

12. SEQUENTIAL UNBLOCKED BINARY FILES WRITTEN BY FTN 3.0 USING 'CALL FTNBIN' MAY BE PROCESSED BY INSERTING

```
FILE(DATA,BT=C,RT=S)
LDSET(FILES=DATA)
```

BEFORE THE 'LGO', WHERE 'DATA' IS THE LFN FOR THE FILE AS IT APPEARS ON THE PROGRAM CARD. ANY FILES WRITTEN BY FTN 4.0 WHICH ARE TO BE USED BY FTN 3.0 PROGRAMS MUST HAVE THE SAME CARDS.

13. FTN BINARY (UNFORMATTED) FILES MAY NOT SET BUFFER SIZE ON THE PROGRAM CARD BELOW 512 WORDS UNLESS SPECIAL FILE TYPE ABOVE. IF IT IS NECESSARY TO USE SMALLER BUFFERS FOR INTERCOM FL LIMITATIONS, THE FOLLOWING CONTROL CARDS WILL BE REQUIRED:

```
FILE(TAPENN,RT=S,BT=C)
XEQ,LDSET=FILES=TAPENN,LOAD=LGO
```

14. FORTRAN READS AND WRITES UTILIZE RECORD MANAGER TO PROCESS ALL I/O REQUESTS. THE USER MAY BYPASS FORTRAN I/O PROCESSING AND CALL RECORD MANAGER ROUTINES DIRECTLY USING 'CALL GET' AND 'CALL PUT'. IN ADDITION, THERE ARE ROUTINES FOR FILE DEFINITION, OPEN, CLOSE, REWIND, FORWARD SPACE, BACKSPACE. THESE ROUTINES ALLOW THE USER TO PROCESS FILES WHICH ARE NOT IN ONE OF THE 3 STANDARD FORTRAN TYPES. (FTN, 8-39; FTN/S, CHAPTER III-6; FTNRM)

15. TO RUN ANY SCOPE 3.3 BINARY OBJECT PROGRAM ON NOS/BE 1.0 (NOT INVOLVING TSKLOAD, PRELOAD, OR ANY SPECIAL DEVICE LIBRARIES SUCH AS SC4020 OR CALCOMP), PRECEDE THE PROGRAM CALL CARD BY

```
ATTACH,SYSMISC.
LDSET(LIB=SYSMISC/SYSIO) OR LIBRARY(SYSMISC,SYSIO)
```

EXAMPLE:

```
JOB CARD
CHARGE CARD
VSN,BINARY=CA9999. (OPTIONAL)
LABEL,BINARY,L=XXXX,R,D=HY,Z,VSN=CA9999,NORING.
ATTACH,SYSMISC.
LIBRARY(SYSMISC,SYSIO) OR LDSET(LIB=SYSMISC/SYSIO)
BINARY.
" 7/8/9 FOR
(DATA)
" 6/7/8/9 EOF
```

16. SCOPE 3.3 OBJECT MODULES MAY NOT BE MIXED WITH SCOPE 3.4 AND/OR NOS/BE MODULES IN A SINGLE PROGRAM. WHEN ANY PART OF THE PROGRAM IS RECOMPILED, EITHER THE ENTIRE PROGRAM MUST BE RECOMPILED IN NOS/BE AND THE SPECIAL LDSET FLIMINATED OR ELSE THE SCOPE 3.3 VERSION OF THE PROPER COMPILER MUST BE ATTACHED FROM PERMANENT FILE. (E.G., ATTACH,FTN,FTN3P0LV340,ID=CSYS.)

17. SORT/MERGE MAY BE EXECUTED DIRECTLY FROM A FORTRAN PROGRAM (FTN, 8-44; FTN/S, CHAPTER III-16). NOTE THAT AN 'RFL' CARD IS REQUIRED FOR EXECUTION.

A PROGRAM WHICH DYNAMICALLY ALLOCATES CORE (INSTEAD OF AN RFL CARD) MUST INSURE THAT BLANK COMMON IS DEFINED LARGE ENOUGH TO HOLD ALL REQUIRED DATA SINCE SORT/MERGE USES ADDITIONAL CORE BEGINNING AT LWA+1 (SEE LOAD MAP).

18. BEGINNING AT LEVEL 401, READMS/WRITMS FILES ARE OF TYPE BT=I, RT=U AND WILL PROCESS FASTER. TO USE READMS FILES CATALOGED PRIOR TO LEVEL 401, 'FILE' AND 'LDSET' CARDS MUST BE SUPPLIED. FOR EXAMPLE,

FILE,LFN,BT=I,RT=W.

LDSET,FILES=LFN.

SCOPE 3.3 READMS/WRITMS FILES ARE CONVERTED BY UTILITY RANCONV (CONV, CHAPTER 8) TO BT=I, RT=W.

IF A READMS PERMANENT FILE IS MODIFIED, AN 'EXTEND' CONTROL CARD MUST BE GIVEN.

19. WHEN 'FTN,ER' IS USED, OBJECT CODE WILL INCLUDE AN EXTRA INSTRUCTION FOR EACH STATEMENT FOR USE BY REPRIEVE, IF JOB ABORTS.

20. ALL ARITHMETIC STATEMENT FUNCTIONS AND INTRINSIC FUNCTIONS ARE IN-LINE CODE, WHETHER TRACE IS ON OR OFF.

*** TIME-SHARING FORTRAN ***

TIME-SHARING MODE FORTRAN (TS ON CONTROL CARD, SEE CCRM, 4-2) IS A ONE-PASS COMPILER. BECAUSE FEWER DISK ACCESSSES ARE REQUIRED, COMPILATION IS FASTER THAN IN OPTIMIZING MODE (OPT=0, 1, 2). OBJECT CODE IS NOT HIGHLY OPTIMIZED AND EXECUTES ABOUT AS FAST AS OPT=0 CODE. FTN,TS REQUIRES CM40000 MINIMUM.

IN ADDITION, TS FORTRAN WILL ACCEPT SOME KEYWORD MISSPELLINGS AND PUNCTUATION ERRORS. WARNING DIAGNOSTICS WILL BE ISSUED SINCE SUCH ERRORS MAY BE FATAL IN OPTIMIZING MODE.

DIAGNOSTIC MESSAGES ARE LISTED AFTER THE LINE WHERE THE ERROR WAS DETECTED. (OPTIMIZING FORTRAN GROUPS THEM AFTER THE COMPLETE PROGRAM LISTING.)

BOTH THE PROGRAM LISTING AND THE CROSS REFERENCE MAPS HAVE FORMATS DIFFERENT FROM THOSE OF OPTIMIZING FORTRAN. (SEE FTN, 11-6, 13-15; FTN/S, CHAPTER III-15)

*** ERROR MESSAGES ***

FORTRAN COMPILATION ERRORS

IMMEDIATELY FOLLOWING THE LISTING OF THE FORTRAN CODE, ERRORS ARE IDENTIFIED BY THE CARD NUMBER ASSIGNED BY THE PROCESSOR (INDICATED ON THE EXTREME LEFT SIDE OF THE LISTING). THE SEVERITY OF THE ERROR IS INDICATED BY THE CODE ACCOMPANYING THE MESSAGE:

I - INFORMATIVE. HAS NO EFFECT ON COMPILATION OR EXECUTION
 FC - FATAL TO COMPILATION
 FE - FATAL TO EXECUTION

FOR EXAMPLE,

| CARD NO. | SEVERITY | DIAGNOSTIC |
|----------|----------|---|
| 9 | FE | THE OPERATOR INDICATED (-, +, *, /, OR **) MUST BE FOLLOWED BY A CONSTANT, NAME, OR LEFT PARENTHESIS. |
| 13 | FE | NO MATCHING RIGHT PARENTHESES. |
| 21 | FE | ILLEGAL CHARACTER. THE REMAINDER OF THIS STATEMENT WILL NOT BE COMPILED. |
| 30 | I | NO END CARD. END LINE ASSUMED. |
| 30 | FE | UNDEFINED STATEMENT NUMBERS, SEE BELOW. (NOTE - THIS ERROR WILL REFERENCE THE END STATEMENT.) |

NOTE THAT FC AND FE COMPILATION ERRORS WILL ABORT THE JOB UNLESS A=0 OPTION WAS USED, IN WHICH CASE THE LOADER WILL PROCEED UNTIL IT ENCOUNTERS AN UNCOMPILED ROUTINE, FLAGGED BY AN ILLEGAL NAME. THE FOLLOWING MESSAGE THEN APPEARS IN THE DAYFILE:

FATAL LOADER ERROR -
 ATTEMPT TO LOAD SUPPRESSED BINARY

FORTRAN OBJECT TIME ERRORS

THE FORTRAN LIBRARY AND IO ROUTINES DETECT MANY OBJECT-TIME ERRORS. SOME ARE INFORMATIVE AND PRODUCE A MESSAGE IN THE OUTPUT LISTING ONLY. OTHERS ARE FATAL, PRODUCING THE MESSAGE "FTN - FATAL ERROR NN" IN THE DAYFILE AND A MORE DETAILED MESSAGE IN THE OUTPUT. THE OPERATING SYSTEM DETECTS CERTAIN OTHER ERRORS, SUCH AS ARITHMETIC MODE ERRORS. LEVEL 420 FTN CONTROL CARD OPTION ER (DEFAULT WITH OPT=0) REPRIEVES THESE AND INTERPRETS THEIR MEANING AND LOCATION IN THE DAYFILE, SUCH AS "INFINITE VALUE IN XYZ NEAR LINE N". FATAL ERRORS CAUSE ALL "OUTPUT" FILE BUFFERS TO BE EMPTIED, BUT NOT OTHER FILE BUFFERS.

SUBROUTINE 'ERRSET' MAY BE UTILIZED TO ALLOW THE PROGRAM TO ACCEPT CERTAIN INPUT DATA ERRORS (CCRM, 4-16; FTN, 8-20; FTN/S, III-3-6). SUBROUTINE 'RECOVR' WITH SUITABLE AUXILIARY ROUTINE MAY BE USED TO GAIN TEMPORARY CONTROL AFTER OTHERWISE FATAL SYSTEM ERRORS (CCRM, 4-16; FTN, 8-12; FTN/S, I-8-13; NOSBE, 6-22; SCOPE 12-20). SUBROUTINE 'SYSTEMC' MAY BE USED TO CHANGE THE TREATMENT OF ERRORS (FTN, 8-15; FTN/S, III-3-2).

JUNE 1977

PAGE 4-15

*** READING DUMPS - FORTRAN DEBUGGING ***

THE DAYFILE IS THE FIRST PLACE TO LOOK FOR ERRORS. SYSTEM ERROR MESSAGES AND MESSAGES FROM THE OPERATOR ARE LISTED THEREIN (SEE FTN, 8-9; FTN/S, III-2-16). EACH CONTROL CARD EXECUTED IS LISTED, AS IS THE TIME.

EXECUTION ERRORS GENERALLY ARE OF THREE TYPES:

- FORTRAN ERRORS - DETECTED IN FORTRAN OBJECT TIME ROUTINES
- NOS/BE ERRORS - DETECTED BY THE SYSTEM
- MODE ERRORS - DETECTED BY THE CPU

(SEE CCRM, 2-21, 4-14)

IF FATAL FORTRAN COMPILER ERRORS ARE FOUND IN ANY SUBPROGRAM, A LINE IS GENERATED IN THE DAYFILE FOR THAT SUBPROGRAM. AN ATTEMPT TO LOAD THE RESULTING INVALID BINARY FILE (I.E., LGO) WILL GIVE A LOADER ERROR, UNLESS "DEBUG" IS IN USE. THE FTN COMPILER GENERATES A REFERENCE MAP FOR EACH ROUTINE, BASED UPON THE VALUE FOR R ON THE FTN CARD. THE DEFAULT, R=1, GIVES A MAP ADEQUATE FOR LOCATING ITEMS IN A DUMP (SEE FTN, CHAPTER 12; FTN/S, CHAPTER III-1).

THE LOADER GENERATES A DIRECTORY (CORE MAP) OF ENTRY POINTS, THE FIRST WORD ADDRESS OF PROGRAM UNITS WITH CROSS REFERENCE, AND THE LOCATION OF COMMON BLOCKS, AS WELL AS A LIST OF UNSATISFIED EXTERNALS (SUBPROGRAMS NOT FOUND). THE TOTAL CORE REQUIRED FOR THE LOADED PROGRAM IS INDICATED. FOR BASIC (NON-SEGLOAD/NON-OVERLAY) LOADS, THE TOTAL CORE REQUIRED TO LOAD THE PROGRAM IS ALSO GIVEN ON THE LAST LINE. LOCATIONS 0-1118 IN THE USER'S FL ARE RESERVED FOR THE SYSTEM COMMUNICATION REGION AND LOADER POINTERS. NOTE THAT EACH I/O BUFFER HAS A DEFAULT SIZE OF 1001 OCTAL (PLUS A 35-WORD FILE INFORMATION TABLE). (SEE SUG, 8-20)

A DMPX IS GENERATED WHENEVER A PROGRAM IS ABORTED BY THE SYSTEM. THIS DUMP INCLUDES THE CONTENTS OF ALL REGISTERS, THE FIRST 100 WORDS IN FL AND 100 WORDS BEFORE AND AFTER THE LOCATION IN THE PROGRAM ADDRESS (P) REGISTER AT THE TIME OF ABORT. THE FATAL ERROR IN A PROGRAM RUN ON THE CDC 6600 MAY ACTUALLY OCCUR SEVERAL INSTRUCTIONS BEFORE THE P ADDRESS DUE TO THE MULTIPLE ARITHMETIC UNITS. IF A RELATIVE DUMP WAS PLANTED AFTER AN EXIT CARD, THIS WILL FOLLOW THE DMPX. SINCE INSTRUCTIONS ARE 15 OR 30 BITS IN LENGTH, EACH CM WORD IS PRINTED IN FOUR COLUMNS FOR EASY READING. (SEE SUG, 8-26)

WHEN PROGRAM STOPS WITH ERROR MODE 2 OR 4, ATTEMPT TO LOCATE INVALID VALUE IN C(AI) LIST. THEN AI IS THE ADDRESS OF THE NUMBER. CALCULATE THE NAME OF THE NUMBER USING CORE MAP AND REFERENCE MAP WITH OCTAL ARITHMETIC. IF C(AI) IS BLANK, THEN AI IS THE OUT OF RANGE ADDRESS IN ERROR MODE 1. TO BACK TRACE WHERE A ROUTINE WAS CALLED, LOOK AT THE 0400XXXXXX WORD WHICH IS USUALLY THE THIRD WORD AFTER THE LOADER MAP ADDRESS OF THE ROUTINE.

JUNE 1977

PAGE 4-16

*** FORTRAN CALLABLE UTILITIES ***

ERRSET 'CALL ERRSET (KOUNT, MAX)' ALLOWS THE USER TO ENCOUNTER
MAX ERRORS IN FORTRAN FORMATTED INPUT DATA BEFORE FATAL
TERMINATION (DEFAULT IS 1). THE CURRENT ERROR COUNT IS
AVAILABLE TO THE USER IN KOUNT. (SEE FTN, 8-20; FTN/S,
I-8-16)

FORM1 MAY BE USED TO CALL THE SYSTEM PROGRAM 'FORM' FROM A
FORM2 FORTRAN PROGRAM. DIRECTIVES ARE PASSED TO 'FORM' EITHER
DIRECTLY OR THRU AN EXTERNAL FILE.
(SEE FORM, 5-4,6,7, 6-14)

RECOVR SHOULD BE USED TO GAIN TEMPORARY CONTROL AFTER AN ABORT
CONDITION TO CLOSE FILES WHICH AN INTERACTIVE INTERCOM JOB
MIGHT OTHERWISE LEAVE OPEN, AND TO CLOSE OUTPUT BUFFERS
FOR FILES NEEDED FOR A LATER JOB, SUCH AS SC4060 OR
CALCOMP (SEE FTN, 8-12; FTN/S, I-8-13; NOSBE, 6-22; SCOPE,
12-20). FOR EXAMPLE,

```

      EXTERNAL MYEND
      CALL RECOVR (MYEND, 778, 0)
WHERE USER SUPPLIES
      SUBROUTINE MYEND (EX, Y, RA)
      DIMENSION EX(17)
C   CLOSE 4020 AND END FILE ALTERNATE OUTPUT
      CALL PLTND (0)
      END FILE 12
      STOP
      END
    
```

SPY

WILL HELP LOCATE AREAS IN A PROGRAM WHERE THE CODE IS INEFFICIENT. SPY WILL RECORD RESULTS ON FILE 'DOSSIER' SHOWING HOW MUCH RELATIVE TIME WAS SPENT IN EACH 'BINW' WORDS FROM 'LOW' TO 'HIGH'.

TO TURN ON SPY:

CALL SPYONF (LOW, HIGH, NAME, BINW)

LOW - START (RELATIVE TO RA) OF AREA TO BE ANALYZED

HIGH - END (RELATIVE TO RA) OF AREA TO BE ANALYZED

NAME - LABEL FOR THIS INVOCATION OF SPY

LEFT-JUSTIFIED, 1-8 CHARACTERS (E.G., 3HABG)

BINW - WIDTH OF SPY ANALYSIS. MAY BE 100B, 40B, 20B, 10B (OR 64, 32, 16, 8).

IN MOST INSTANCES, 100B (64) IS SUFFICIENT.

TO TURN OFF SPY:

CALL SPYOFF

BEFORE PROGRAM TERMINATES, IT MUST "CALL SPYOFF". MAY NEED "RECOVR" (SEE CCRM, 4-15).

EXAMPLE:

```
...  
CALL SPYONF (101B, 40000B, 6HMYPROG, 100B)  
...  
CALL SPYOFF  
...
```

NOTE: WHEN USING SPY ON THE 6700, CPB SHOULD NOT BE SPECIFIED ON THE JOB CARD.

(SEE CCRM, 2-7: PRNTSPY, FOR DIRECTIONS ON PRINTING THE RESULTING SPY ANALYSIS. SEE ALSO CONV, CHAPTER 11.)

FOR ADDITIONAL FTN-CALLABLE CONTROL CARD ROUTINES SUCH AS ROUTE, REQUEST, SKIPFIL, UNLOAD, ZPFUNC (ATTACH, CATALOG, ETC.), SEE CCRM, 10-4: NSRDC: CCLIB/N.

*** SAMPLE FORTRAN DECK SETUPS ***

COMPILE AND LOAD ONLY

| | |
|----------------------------|---------------------------------------|
| XXXX,CM54000. | NAME/CODE |
| CHARGE,XXXX,JJJJJJJJJJ. | |
| FTN. | ** SEE OPTIONS ON PAGE 4-1 |
| LOAD(LGO) | ** CHECKS FOR MISSING SUBROUTINES |
| NOGO. | ** CAUSES LOADING MAP TO BE GENERATED |
| ' 7/8/9 EOR | |
| (ALL FORTRAN SOURCE DECKS) | |
| " 6/7/8/9 EOF | |

COMPILE AND EXECUTE (NO BINARY DECKS)

| | |
|----------------------------|---------------------|
| JORNAME,..... | NAME/CODE |
| CHARGE,..... | |
| FTN. | |
| LGO. | ** LOAD AND EXECUTE |
| ' 7/8/9 EOR | |
| (ALL FORTRAN SOURCE DECKS) | |
| ' 7/8/9 FOR | |
| (DATA DECK) | |
| " 6/7/8/9 EOF | |

COMPILE AND EXECUTE, CATALOG BINARY PROGRAM

| | |
|---|---------------------------------|
| JORNAME,..... | NAME/CODE |
| CHARGE,..... | |
| REQUEST,BINARY,*PF. | ** GET SPACE FOR PERMANENT FILE |
| FTN(B=BINARY,OPT=1) | ** OBJECT CODE TO FILE 'BINARY' |
| MAP(CN) | ** FULL MAP FOR DOCUMENTATION |
| BINARY. | ** LOAD FROM FILE AND EXECUTE |
| CATALOG,BINARY,PROGBIN,ID=XXXX,XR=RDONLY. | ** MAKE FILE PERMANENT |
| ' 7/8/9 EOR | |
| (ALL FORTRAN SOURCE DECKS) | |
| ' 7/8/9 EOR | |
| (DATA DECK) | |
| " 6/7/8/9 EOF | |

COMPILE, LOAD AND EXECUTE (WITH PUNCHB OPTION)

```

JOBNAME,..... NAME/CODE
CHARGE,.....
FTN (B=PUNCHB,OPT=1) ** SEE OPTIONS ABOVE
LOAD (INPUT) ** LOAD BINARY DECKS INTO MEMORY
PUNCHB. ** LOAD (FORTRAN COMPILATION) FROM PUNCHB
** FILE AND EXECUTE

* 7/8/9 EOR
  (ALL FORTRAN SOURCE DECKS)
* 7/8/9 EOR
  (ALL OTHER BINARY DECKS)
* 7/8/9 EOR (PART OF BINARY DECK)
* 7/8/9 FOR (NOT NEEDED IF NO DATA)
  (DATA DECK)
" 6/7/8/9 EOF

```

EXECUTE ONLY (FROM BINARY PROGRAM DECK)

```

JOBNAME,..... NAME/CODE
CHARGE,.....
INPUT. ** LOAD AND EXECUTE PROGRAM
* 7/8/9 EOR
  (ALL BINARY DECKS)
* 7/8/9 EOR (PART OF BINARY DECK)
* 7/8/9 FOR (NOT NEEDED IF NO DATA)
  (DATA DECK)
" 6/7/8/9 EOF

```

EXECUTE ONLY (BINARY PROGRAM ON PERMANENT FILE)

```

JOBNAME,..... NAME/CODE
CHARGE,.....
ATTACH (OBJECT,PROGBIN,ID=XXXX,MR=1) ** RETRIEVE PERMANENT FILE
OBJECT. ** LOAD FILE AND EXECUTE
* 7/8/9 EOR (NOT NEEDED IF NO DATA)
  (DATA DECK)
" 6/7/8/9 EOF

```

COMPILE AND EXECUTE (TAPE USED)
(DUMP, IF EXECUTION ERRORS)

THIS JOB NEEDS TWO TAPES: TAPE27, A HIGH-DENSITY BINARY OUTPUT TAPE
AND TAPE8, A HIGH-DENSITY, FORMATTED STRANGER SLOT TAPE. BINARY INPUT
DATA ON DISK FILE TAPE16.

```

JOBNAME,MT2,.....                     NAME/CODE
CHARGE,XXXX,JJJJJJJJJJ.
FTN.
ATTACH,TAPE16,ID=XXXX.
VSN,TAPE8=SLOT22=FOREGN.
LABEL,TAPE27,L=XXXXDATA1,D=HI,W,VSN=CA8888,RING.
REQUEST,TAPE8,HI,S,RING.
LGO.
RETURN,TAPE16,TAPE27,TAPE8.
EXIT.                                  ** FOLLOWING EXECUTED ONLY AFTER ERROR
DMP,46000.                             ** DUMP FIRST 46000 OF FIELD LENGTH
RETURN,TAPE16,TAPE27,TAPE8.
* 7/8/9      EOR
  PROGRAM LINE (INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE27,
  1 TAPE16,TAPE8)
  ...
  READ (16) (A(I),I=1,32)              ** - SEE BELOW
  ...
  WRITE (8,5) (B(I),I=1,32)            ** - SEE BELOW
  WRITE (27) (B(I),I=1,32)             ** - SEE BELOW
  ...
  END
* 7/8/9      EOR (NOT NEEDED IF NO DATA)
(DATA DECK)
** 6/7/8/9   EOF

** - PROGRAM WOULD BE MORE EFFICIENT, ASSUMING 'DIMENSION A(32), B(32)'
  IS USED, IF THE I/O LISTS USED JUST THE ARRAY NAME (E.G.,
  'READ (16) A').

```

*** SAMPLE FORTRAN INPUT/OUTPUT ***

ENCODE, BUFFEROUT

```

PROGRAM TSTAPE (TAPE9, ...)
C   TO WRITE A STRANGER TAPE FOR AN OFF-LINE DEVICE MAY USE
C   BUFFER OUT (WHICH MEANS THE DATA MUST BE CONTIGUOUS IN CORE).
C   ALTERNATIVELY, 'FILE' CARDS AND STANDARD FORTRAN READS AND
C   WRITES MAY BE USED.
    DIMENSION OUTBUF(8), B(50, 20)
    DO 7 NI=1,N
      ...
      ...
      ENCODE(80, 15, OUTBUF) I, J, B(I, J), C, D, E, NI
15  FORMAT(2I3, F12.5, 3F12.6, 18X, 4HCARD, I4)
C   CREATE A STRANGER TAPE CODED FILE (EVEN PARITY)
    BUFFER OUT(9, 0) (OUTBUF(1), OUTBUF(8))
C   WAIT UNTIL DATA IS WRITTEN
    IF (UNIT(9) .GE. 0) GO TO 8
    END FILE 9
    REWIND 9
    STOP
8   ... PRINT MSG FOR PARITY ON WRITE
END

```

BUFFERIN, DECODE

```

PROGRAM PATRICK(TAPE2=0, OUTPUT=128, INPUT=128)
C   EXAMPLE OF USING BUFFERIN TO SURVEY THE CONTENTS OF A
C   FILE WRITTEN WITH BT=C, RT=S (SEE CCRM, 4-11)
    DIMENSION A(15000)
    DATA I, M/ 0, 1/
5   BUFFER IN (2, 1) (A(1), A(15000))
C   WAIT UNTIL DATA IS FINISHED READING THEN TO GO 10
C   CHECK FOR EOF OR PARITY AND GO TO 12 OR 14 RESPECTIVELY
    IF(UNIT(2)) 10, 12, 14
14  ... CODE TO PRINT PARITY MSG, CONTINUE PROCESSING AT STMT 10
10  L=LENGTH(2)
    I=I+1
C   PRINT RECORD NUMBER, LENGTH, FIRST 4 WORDS
C   IF RECORD EXCEEDS 15000 WORDS, LENGTH IS TRUNCATED
    PRINT 6, I, L, (A(K), K=1, 4)
6   FORMAT (" RECORD " I4, " LENGTH " I6, 3X, 4022)
C   DECODE SELECTED FIELDS FROM ALPHA TO NUMERIC AND PRINT
    DECODE (110, 16, A(1)) J, B, C
16  FORMAT (47X, I3, 2(20X, F10.6))
C   CHARACTER COUNT FOR SINGLE DECODE CANNOT EXCEED 150
    DECODE(110, 16, A(12)) N, D, E
    PRINT 18, J, N, B, C, D, E
18  FORMAT(" KEY FIELDS ", 2I6, 4F20.6)
    GO TO 5
12  PRINT 7, M
7   FORMAT (" END OF FILE " I4)
C   IF TAPE HAS MORE THAN ONE FILE, ADD CODE TO
C   INCREMENT M AND CONTINUE PROCESSING TAPE
END

```

*** OVERLAY ***

(SEE LOADER, CHAPTER 4 OR FTN, 7-19 OR FTN/S, CHAPTER I-12)

OVERLAY ALLOWS A USER TO FIT A LARGE PROGRAM INTO SMALLER AMOUNT OF CORE. EACH OVERLAY CONTAINS A MAIN PROGRAM (AND OPTIONALLY SUBPROGRAMS) AND IS A SELF-CONTAINED ABSOLUTE PROGRAM LOADED AND EXECUTED WITHOUT LINKING DURING EXECUTION. OVERLAYS MAY BE PRIMARY OR SECONDARY, ALLOWING ONLY SELF OR UPWARD REFERENCES. DATA COMMUNICATION BETWEEN OVERLAYS IS ONLY THROUGH BLANK OR LABELLED COMMON.

OVERLAY DIRECTIVES BEGIN IN COLUMN 7. THE MAIN PROGRAM IN EACH LEVEL TRANSFERS CONTROL BACK TO THE CALLER WHEN THE 'END' CARD IS EXECUTED. THE MAIN PROGRAM FOR EACH LEVEL MAY BE A SHORT ONE, PERHAPS CREATED TO CALL OTHER SUBPROGRAMS IN THE LEVEL, OR A FORMER SUBPROGRAM MAY BE MODIFIED. THE LOADER REQUIRES AT LEAST 4000 (OCTAL) LOCATIONS FOR LOADING THE (0,0) OVERLAY.

OVERLAY DIRECTIVES
(PART OF FTN SOURCE PROGRAM)

OVERLAY (LFN,I,J) PRECEDES MAIN PROGRAM OF EACH OVERLAY
WHERE LFN IS OVERLAY FILE NAME
I IS PRIMARY LEVEL (0-77 OCTAL)
J IS SECONDARY LEVEL (0-77 OCTAL)

CALL OVERLAY(LFNLOC,I,J) CALL INTO CORE AND EXECUTE
WHERE LFNLOC MAY BE LEFT DISPLAY CODE
OF LFN OR A VARIABLE CONTAINING THE LFN
IN LEFT DISPLAY CODE, AND I,J ARE 1-63
DECIMAL.
FOR EXAMPLE, CALL OVERLAY(5LPROG1,1,0)
(SEE CCRM, 7-16 FOR USE IN A PROGRAM IN A
LIBRARY.)

PROGRAM NAME I NO ARGUMENT LIST ON SUBLEVEL PROGRAM.
ALL FILES MUST BE IDENTIFIED ON PROGRAM
CARD IN OVERLAY (0,0)

NOTE: CALLING IN A SECONDARY OVERLAY WILL NOT AUTOMATICALLY LOAD
THE APPROPRIATE PRIMARY OVERLAY.

*** SEGMENTATION ***

SCOPE 3.3 SEGMENTATION IS NO LONGER AVAILABLE.

SEGMENTATION IN NOS/BE IS RECOMMENDED AND SHOULD BE USED INSTEAD OF OVERLAY. SEE CCRM, 2-19.

SEGMENTATION ALLOWS THE USER TO FIT A LARGE PROGRAM INTO A SMALL AMOUNT OF CORE. THE PROGRAM CONSISTS OF ONE MAIN PROGRAM AND SEVERAL SUBPROGRAMS. BOTH UPWARD AND DOWNWARD REFERENCES ARE PERMITTED. MANY LEVELS, EACH CONTAINING MANY SEGMENTS ARE ALLOWED, RESTRICTED ONLY TO A TOTAL OF 4093 SEGMENTS. MOST EFFICIENT USE OF CORE CAN BE OBTAINED BY USING ONLY ONE LEVEL AND NO BLANK COMMON. THIS WAY, CORE WILL BE ALLOCATED DYNAMICALLY. BECAUSE THE STATEMENTS WHICH DEFINE THE SEGMENT STRUCTURE ARE NOT A PART OF THE PROGRAM, NO SOURCE CODE CHANGES ARE REQUIRED TO RESTRUCTURE THE PROGRAM.

*** SETTING UP OVERLAY ***

ORIGINAL PROGRAM

```
C      PROGRAM TESOS(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE14)
      COMMON /NAMED/ X,Y,Z(10,100)
      TESOS CALLS TWO INDEPENDENT SUBROUTINES  S GOOD  11/23/70
      L=1
      READ(5,9) X,Y
      ...
      CALL BAB
      ...
      F=X+Y
      CALL CT(F,L,Q)
      ...
      STOP
9  FORMAT(2F12.6)
      END

      SUBROUTINE BAB
      DIMENSION H(75,20)
      COMMON /NAMED/ X,Y,Z(10,100)
      ...
      DO 6 K=1,5
      ...
      DO 6 I=1,10
      WRITE(14) (Z(I,J),J=1,100)
6  CONTINUE
      REWIND 14
      RETURN
      END

      SUBROUTINE CT (F,L,Q)
      DIMENSION P(50,100)
      IF(L.EQ. 1) GO TO 24
      ...
      Q=P(L,L)
      RETURN
24 DO 5 I=1,50
      5 READ(14) (P(I,J), J=1,100)
      ...
      RETURN
      END
```

AD-A043 559

DAVID W TAYLOR NAVAL SHIP RESEARCH AND DEVELOPMENT CE--ETC F/G 9/2
COMPUTER CENTER REFERENCE MANUAL, REVISION.(U)
JUN 77 S E GOOD, D V SOMMER
CMLD-77-11 NL

UNCLASSIFIED

2 OF 3

AD
A043 559

NL

*** SAMPLE OVERLAY ***
 (SEE ALSO CCRM, 7-16)

```

OVERLAY(TESOS,0,0)
PROGRAM TESOS(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE14)
C TESOS CALLS TWO INDEPENDENT SUBROUTINES S GOOD 11/23/70
C ** ENDPUNCHING INDICATES CHANGES IN ORIGINAL PROGRAM
COMMON /NAMED/ X,Y,Z(10,100)
COMMON /CCOM/ F,L,Q
L=1
READ(5,9) X,Y
...
CALL OVERLAY (5HTESOS,1,0)
...
F=X+Y
CALL OVERLAY (5HTESOS,2,0)
...
STOP
FORMAT(2F12.6)
END

OVERLAY(TESOS,1,0)
PROGRAM BACA
C OVERLAY BY DUMMY MAIN PROGRAM
CALL BAB
END
SUBROUTINE BAR
DIMENSION H(75,20)
COMMON /NAMED/ X,Y,Z(10,100)
...
DO 6 K=1,5
...
DO 6 I=1,10
WRITE(14) (Z(I,J),J=1,100)
6 CONTINUE
REWIND 14
RETURN
END

OVERLAY(2,0)
PROGRAM CTCA
C OVERLAY BY MODIFY FORMER SUBPROGRAM
COMMON /CCOM/ F,L,Q
DIMENSION P(50,100)
IF(L.EQ. 1) GO TO 24
...
Q=P(L,L)
GO TO 40
24 DO 5 I=1,50
5 READ(14) (P(I,J), J=1,100)
...
40 CONTINUE
END

```

*** COMPILE, LOAD AND CATALOG ABSOLUTE PROGRAM ***

SIMPLE LOAD

| JOBNAME,..... | NAME/CODE |
|----------------------------------|--------------------------------|
| CHARGE,..... | |
| FTN. | |
| REQUEST,MYPROG,*PF. | ** MYPROG FOR ABSOLUTE MODULE |
| LOAD,LGO. | |
| NOGO,MYPROG. | ** ABSOLUTE MODULE ONTO MYPROG |
| CATALOG,MYPROG,ID=XXXX,XR=RONLY. | |
| * 7/8/9 EOR | |
| PROGRAM TEST (...) | |
| ... | |
| " 6/7/8/9 EOF | |

OVERLAY LOAD

| JOBNAME,..... | NAME/CODE |
|---|---------------------------------|
| CHARGE,..... | |
| FTN. | |
| REQUEST,MYOVLY,*PF. | ** MYOVLY FOR ABSOLUTE OVERLAY |
| LOAD,LGO. | |
| NOGO. | ** ABSOLUTE OVERLAY ONTO MYOVLY |
| CATALOG,MYOVLY,ID=XXXX,XR=RONLY. | |
| * 7/8/9 EOR | |
| OVERLAY (MYOVLY, 0, 0) | |
| C 'MYOVLY' IS THE NAME OF THE ABSOLUTE OVERLAY FILE | |
| PROGRAM TEST (...) | |
| ... | |
| " 6/7/8/9 EOF | |

SEGLOAD

| JOBNAME,..... | NAME/CODE |
|----------------------------------|--|
| CHARGE,..... | |
| REQUEST,LGO,*PF. | |
| FTN. | |
| CATALOG,LGO,MYSEG LGO,ID=XXXX. | ** SAVE RELOCATABLE MODULES FOR POSSIBLE RE-SEGMENTATION |
| REQUEST,MYSEGL,*PF. | ** MYSEGL FOR ABSOLUTE SEGMENTS |
| SEGLOAD,B=MYSEGL. | |
| LOAD,LGO. | |
| NOGO. | ** ABSOLUTE SEGMENTS ONTO MYSEGL |
| CATALOG,MYSEGL,ID=XXXX,XR=RONLY. | |
| * 7/8/9 EOR | |
| (FTN SOURCE PROGRAM) | |
| * 7/8/9 EOR | |
| (SEGLOAD CONTROL CARDS) | |
| " 6/7/8/9 EOF | |

*** MINNESOTA FORTRAN (MNF) ***

THE MINNESOTA FORTRAN COMPILER (MNF), WRITTEN AT THE UNIVERSITY OF MINNESOTA, IS AVAILABLE ON THE DTNSRDC 6000 COMPUTERS. LOCAL CONTACT IS NCRA TAYLOR, CODE 189.1, TELEPHONE (202) 227-1907.

THE MNF COMPILER MAY BE ACCESSED AS FOLLOWS:

```

JOBNAME,CM53000,....      NAME / CODE
CHARGE,....
ATTACH,MNFTN.
LIBRARY,MNFTN.
MNF,<OPTIONS>.
'  7/8/9      EOR
  (FORTRAN SOURCE PROGRAM)
'  7/8/9      EOR
  (DATA CARDS, IF ANY)
"  6/7/8/9    EOF

```

SOME OF THE <OPTIONS> ARE:

I=LFN READ SOURCE CARDS FROM FILE 'LFN' (DO NOT USE 'I' ALONE).
 B COMPILE ONLY, DO NOT LOAD-AND-GO. PUT BINARY OUTPUT ON 'LGO'.
 B=LFN COMPILE ONLY, DO NOT LOAD-AND-GO. PUT BINARY OUTPUT ON 'LFN'.
 L=LFN PUT PROGRAM AND CROSS REFERENCE LISTINGS ON FILE 'LFN'.
 T TURN ON 6 OBJECT-TIME TRACING FEATURES (INCLUDING SUBSCRIPT BOUND CHECKING) WITH NO CHANGE TO SOURCE CODE.
 P=NNNN MAXIMUM (DECIMAL) NUMBER OF LINES ON FILE 'OUTPUT' (INCLUDES ANY TRACE OUTPUT). (DEFAULT: 5000)

EXAMPLES: MNF(I=COMPILE,T,P=1500)

```

MNF.      (SAME AS MNF,I=INPUT,L=OUTPUT,P=5000.)
          BINARY OBJECT CODE IS PUT ON LGO. PROGRAM IS LOADED
          WITH UNDEFINED CORE SET TO NEGATIVE INDEFINITES WITH
          ADDRESSES, AND EXECUTED. NOTE THAT NO 'LGO.' CARD IS
          NEEDED.

```

THE MNF COMPILER PERFORMS SOME CONSISTENCY CHECKS AMONG RELATED PROGRAMS AND SUBPROGRAMS. THEREFORE, IT IS USEFUL TO COMPILE RELATED ROUTINES TOGETHER WHENEVER POSSIBLE.

MNF HAS THE SAME EXECUTION-TIME ERRORS AS FTN (IMPROPER ARGUMENTS FOR MATH ROUTINES, ETC.), BUT, UNLIKE FTN, THEY ARE ALL FATAL.

COMPATIBILITY WITH FTN:

MNF IS A DIFFERENT COMPILER FROM CDC FORTRAN EXTENDED (FTN), BUT IT COMPILES BASICALLY THE SAME LANGUAGE. THE CDC FTN VERSION 4 MANUAL MAY BE USED, IN GENERAL, AS THE REFERENCE MANUAL FOR THINGS OTHER THAN THE CONTROL CARD OPTIONS. A PROGRAM COMPILED BY MNF MAY CALL SUBPROGRAMS COMPILED BY FTN, FOR EXAMPLE, THE MATH SUBROUTINE LIBRARIES SUCH AS IMSL.

ADVANTAGES:

MNF COMPILES FASTER THAN FTN OPT=0, HAS MORE AND BETTER DIAGNOSTICS, BETTER COMPILATION AND CROSS REFERENCE INFORMATION, MORE POWERFUL DEBUGGING AND TRACING CAPABILITY. IT ANTICIPATES SOME FEATURES OF THE PROPOSED ANSI FORTRAN 77 STANDARD, SUCH AS 'PARAMETER' AND 'IF-THEN-ELSE'. IT ALLOWS 'END=' AND 'ERR=' IN READ STATEMENTS.

DISADVANTAGES:

EXECUTION TIME OF COMPILED PROGRAMS IS SLOWER THAN FTN OPT=1 OR 2. THEREFORE, AFTER A PROGRAM IS DEBUGGED, IT SHOULD BE RECOMPILED AT OPT=1 FOR NORMAL USAGE.

A DOCUMENT (8 PAGES AT 8 LINES PER INCH) DESCRIBING MNF MAY BE OBTAINED BY:

BEGIN,UTILITY,,PROGDOC,OTHER,,MNF,OUTPUT.

*** RATIONAL FORTRAN (RATFOR) ***

RATIONAL FORTRAN IS A PROGRAMMING LANGUAGE THAT HAS THE STRUCTURE FORMING STATEMENTS THAT ALLOW "TOP DOWN" AND "GO TO-LESS" PROGRAMMING. RATFOR IS USED AS A SOFTWARE ENGINEERING TOOL TO AID IN PRODUCING AND MAINTAINING COMPLICATED COMPUTER PROGRAMS. RATFOR IS A PRE-COMPILER (FTN MUST BE EXECUTED AFTER RATFOR). RATFOR IS CLOSER TO THE PROPOSED ANSI FORTRAN 77 STANDARD THAN FTN.

RATFOR IS MAINTAINED ON THE DTNSRDC CDC 6000'S BY MEL HAAS, CODE 1843, (202) 227-1933. QUESTIONS, SUGGESTIONS AND TROUBLE REPORTS SHOULD BE DIRECTED TO HIM.

A DOCUMENT (12 PAGES) DESCRIBING RATFOR MAY BE OBTAINED BY

BEGIN,UTILITY,,PROGDOC,OTHER,,RATFOR,OUTPUT.

***** COBOL *****

*** INTRODUCTION TO COBOL 4 ***

THE COBOL COMPILER IN USE AT DTNSRDC ON THE CONTROL DATA 6000 NOS/BE SYSTEM IS COBOL 4. THE COMPILER IS DESIGNED TO BE A SUPERSET OF ANSI COBOL (1968 STANDARD). A CONTROL CARD OPTION CAUSES THE COMPILER TO DIAGNOSE AS ERRORS ANY NON-ANSI FEATURES USED. THIS CHAPTER DOES NOT DESCRIBE THE COBOL 4 LANGUAGE (SEE PAGE I: COBOL).

*** COBOL CONTROL CARD PARAMETERS ***

EXAMPLES COBOL. DEFAULTS TO I=INPUT,L=OUTPUT,B=LGO
 COBOL(I=COMPILE,LR,B=0)
 COBOL,I=COMPILE,LRM,U.
 COBOL(F=MAIN,OB=OVER,SUB)

| <u>OPTION</u> | <u>ACTION</u> |
|---------------|---|
| B=PUNCHB | GIVES PUNCHED BINARY DECKS OF ALL ROUTINES; NO BCD SEQUENCING AT END OF CARDS |
| B=LFN | PUTS BINARY IMAGES ON FILE 'LFN' |
| B=0 | SPECIFIES NO BINARY OUTPUT |
| B | DEFAULTS TO B=LGO |
| D | SETS A FLAG THAT INHIBITS LOADING OF THE RELOCATABLE BINARY WHEN AN E DIAGNOSTIC IS ENCOUNTERED AND DISPLAYS THE FOLLOWING MESSAGE ON THE DAYFILE: FATAL COBOL ERROR OR D OPTION IN EFFECT |
| DB | CAUSES CODE TO BE GENERATED THAT CHECKS THE UPPER LIMITS OF THE SUBSCRIPTS TO INSURE THAT THEY DO NOT EXCEED THE SIZE OF THE OCCURS CLAUSE. NO SUBSCRIPT CHECKING IS DONE IF DB IS NOT SELECTED. THIS OPTION SHOULD BE USED AS A DEBUGGING AID. |
| DB1 | ALLOWS THE GENERATION OF OBJECT CODE WHICH CALLS THE TRACE FEATURE TO TRACE THE FLOW OF THE PROGRAM. THIS PARAMETER MUST NOT BE SELECTED UNLESS THE TRACE FEATURE IS ENTERED BY THE COBOL SOURCE PROGRAM. SEE COBOL, CHAPTER II-14. |
| E=PROG-NAME | MUST BE USED WHEN THE OUTPUT OF A COBOL COMPILATION IS TO BE ADDED TO A USER LIBRARY WITH EDITLIB. THE PROG-NAME IS THE NAME BY WHICH THE PROGRAM WILL BE CALLED WHEN EXECUTED FROM THE LIBRARY. IT MUST BE FIVE CHARACTERS OR LESS AND MUST NOT DUPLICATE THE NAME IN ANY PROGRAM-ID CLAUSE, ANY ENTRY CLAUSE OR ANY IMPLEMENTOR NAME IN A SELECT OR SPECIAL-NAMES CLAUSE. A LOAD, NOGO SEQUENCE WHICH CREATES AN ABSOLUTE FILE ON A FILE NAMED 'COBCODE' MUST PRECEDE THE EDITLIB DIRECTIVES. |
| I=LFN I | USED IF COBOL SOURCE INPUT IS ON 'LFN' INSTEAD OF 'INPUT' DEFAULTS TO I=INPUT |

L=LFN INDICATES THE FILE ON WHICH A NORMAL LISTING IS WRITTEN. A NORMAL LISTING INCLUDES SOURCE STATEMENTS AND MAJOR (C AND E) DIAGNOSTICS

L=0 LIST OUTPUT IS SUPPRESSED EXCEPT FOR C AND E DIAGNOSTICS

L DEFAULTS TO L=OUTPUT

THE LIST PARAMETER KEY LETTER 'L' MAY BE SUFFIXED BY ANY COMBINATION OF THE FOLLOWING OPTIONS TO PROVIDE FEATURES IN ADDITION TO THE NORMAL LISTING.

X INCLUDES T AND U DIAGNOSTICS ON LIST OUTPUT

R PRODUCES A DATA-NAME AND PROCEDURE-NAME CROSS REFERENCE

C INCLUDES SOURCE STATEMENTS COPIED FROM A SOURCE LIBRARY IN THE LIST OUTPUT

O PRODUCES OBJECT CODE LISTING (USE ONLY WHEN REQUESTED BY 1892)

M PRODUCES A DATA MAP

N OR P CAUSES NON-ANSI FEATURES TO BE DIAGNOSED BY THE COMPILER AS E TYPE ERRORS

OB=LFN BINARY OUTPUT FROM SEGMENTS IS PLACED ON FILE 'LFN'

SUB SUPPRESSES CERTAIN OUTPUT FROM COMPILATION OF A SUBPROGRAM

SUBM INDICATES THAT THIS COBOL SUBPROGRAM WILL BE CALLED FROM A MAIN PROGRAM WRITTEN IN ANOTHER LANGUAGE

U SELECTS THE STANDARD CDC COLLATING SEQUENCE FOR SORT VERB EXECUTION AND INDEXED SEQUENTIAL FILE CREATION INSTEAD OF THE DEFAULT ASCII COLLATING SEQUENCE.

Z IGNORES THE COBOL 4 FEATURES AND PRODUCES AN OBJECT PROGRAM THAT IS COMPATIBLE WITH COBOL 3 FILES.

FOR FURTHER EXPLANATION AND OTHER PARAMETERS, SEE COBOL, CHAPTER II-10.

*** COBOL COMPILATION ERRORS ***

WHEN THE COMPILER DETECTS A SOURCE LANGUAGE ERROR, IT PRINTS OUT ONE OR MORE DIAGNOSTIC MESSAGES FOLLOWING THE SOURCE LISTING. EACH MESSAGE IS ASSOCIATED WITH AN IDENTIFYING NUMBER, A SEVERITY CODE AND THE LINE NUMBER IN WHICH THE ERROR OCCURRED. THE SEVERITY OF THE DIAGNOSTIC IS INDICATED BY A CODE LETTER: T - TRIVIAL; U - UNCONVENTIONAL; E - ERROR; AND C - CATASTROPHIC.

A DAYFILE MESSAGE GIVING THE TOTAL NUMBER OF DIAGNOSTICS IS PRINTED FOR EACH COMPILATION.

*** CONVERSION TO COBOL FROM IBM S/360 ***

THE IBM S/360 COBOL REFERENCED IS COBOL F, HOWEVER THE CONCEPTS APPLY IN MOST CASES TO ANY OTHER VERSION OF COBOL.

1. TO ENCLOSE NON-NUMERIC LITERALS, CDC COBOL USES DOUBLE QUOTES (') WHILE IBM USES SINGLE QUOTE (').

2. ALL DATA PASSED FROM COBOL MAIN PROGRAM TO COBOL SUBPROGRAMS MUST BE DEFINED IN THE COMMON-STORAGE SECTION. IT IS NOT POSSIBLE TO PASS DATA THAT IS DEFINED IN THE WORKING-STORAGE SECTION TO COBOL SUBPROGRAMS. NON-COBOL SUBPROGRAMS CAN ACCEPT ITEMS FROM WORKING STORAGE VIA THE 'USING' PART OF THE CALL STATEMENT. ALL DATA ELEMENTS IN THE COMMON MUST BE DEFINED IN THE SAME ORDER AND BE THE SAME LENGTH IN BOTH MAIN AND SUBPROGRAM.

FOR EXAMPLE IN THE MAIN PROGRAM

```
01 GROUP-DATA-ELEMENT.  
   02 FILLER          PICTURE XXX.  
   02 ELEMENTRY-1     PICTURE X(4).  
   02 ELEMENTRY-2     PICTURE XXX.
```

WHILE IN THE SUBPROGRAM

```
01 GROUP-DATA-ELEMENT PICTURE X(10).
```

WHEN COMPILING A COBOL SUBPROGRAM TO BE LOADED WITH A COBOL MAIN PROGRAM, AN ADDITIONAL OPTION 'SUB' MUST BE INCLUDED ON THE COBOL CONTROL CARD TO PREVENT THE COMPILER FROM WRITING DUPLICATE DATA DIVISION ENTRIES ON THE LGO FILE. ALL FILES USED BY A SUBPROGRAM MUST BE DEFINED IN THE MAIN COBOL PROGRAM.

3. IN THE FILE CONTROL AREA THE SELECT CLAUSE IS USED TO RELATE A LENGTHY COBOL REFERENCE TO A FILE NAME THAT FITS THE CONVENTIONS ESTABLISHED BY THE COMPUTER MANUFACTURER. IN CDC SYSTEMS THE 'IMPLEMENTOR NAME' MUST NOT EXCEED 7 CHARACTERS, MAY NOT BE BOUND BY QUOTES, AND MUST BE THE ACTUAL LOCAL FILE NAME (LFN). CDC CONVENTIONS ARE SIMPLE AND EASY TO USE. THERE ARE FOUR SPECIAL 'IMPLEMENTOR NAMES' INPUT, OUTPUT, PUNCH, AND PUNCHB. FOR EXAMPLE -

```
SELECT CARD-INPUT,      ASSIGN INPUT.  
SELECT PRINTER,         ASSIGN OUTPUT.  
SELECT CARD-PUNCH,      ASSIGN PUNCH.  
SELECT CARD-PUNCH-BIN,  ASSIGN PUNCHB.
```

ALL RECORDS READ FROM CARD-INPUT WILL BE THE CARD IMAGES FROM THE CARD READER. ALL RECORDS WRITTEN TO PRINTER WILL BE PRINTED ON ANY AVAILABLE PRINTER AFTER JOB TERMINATION. ALL RECORDS WRITTEN TO CARD-PUNCH OR CARD-PUNCH-BIN WILL BE PUNCHED AFTER JOB TERMINATION.

3. CONTINUED.

IN IBM COBOL ALL FILES 'SELECT'ED MUST BE ON DD CONTROL CARDS EVEN IF EMPTY AND USED ONLY DURING PROGRAM CHECK-OUT RUNS. WITH CDC 6000 EXTERNAL CONTROL CARDS ARE NOT ALWAYS REQUIRED. IF A NAMED INPUT FILE NEED NOT BE PRESENT WHEN THE OBJECT PROGRAM IS EXECUTED, THE WORD OPTIONAL MUST BE SPECIFIED BETWEEN THE FILE NAME AND THE WORD SELECT. WHEN AN OPTIONAL FILE IS NOT PRESENT, THE FIRST READ ATTEMPT WILL IMMEDIATELY SATISFY THE 'AT END' CONDITION AND CONTROL WILL PASS TO THE PROCEDURE TO BE EXECUTED 'AT END'. IF ONE OR MORE 'SELECT'ED FILES ARE TO BE WRITTEN, THEY WILL BE CREATED BY THE SYSTEM ON MASS STORAGE. ALL DISK OUTPUT FILES, EXCEPT THE SPECIAL FILES OUTPUT, PUNCH, AND PUNCHB WILL BE RELEASED BY THE SYSTEM UPON JOB TERMINATION UNLESS PROVISION IS MADE TO CATALOG THEM.

4. AS IN IBM COBOL AN SD (SORT DESCRIPTION) IS REQUIRED WHEN THE SORT VFRB IS TO BE USED. IN ADDITION, CDC COBOL REQUIRES A SELECT CLAUSE FOR THE OBJECT OF THE SD ENTRY.

FOR EXAMPLE -

SELECT SORT-FILE, ASSIGN TEMP.

SORT-FILE, DATA RECORD IS SORT-RECORD.

SORT-RECORD.

02 DATA-ELEMENT-1 PICTURE XX.

ETC.

5. IN THE FILE DESCRIPTION ENTRIES 'RECORDING MODE' MAY NOT BE F, U, OR V. THE OPTIONS ARE DECIMAL OR BINARY. THE USE OF BINARY RESULTS IN GREATER EFFICIENCY WHEN WRITING NOS/BE TAPES WHICH HAVE A FORMAT PECULIAR TO THE CDC 6000. IF THE 'RECORDING MODE' CLAUSE IS NOT USED, THE COMPILER DEFAULTS TO DECIMAL. TAPES WRITTEN AS DECIMAL AND READ AS BINARY WILL CAUSE TAPE READ ERRORS (AND VICE VERSA). THE 'RECORDING MODE' CLAUSE HAS NO MEANING WHEN THE FILE IS A DISK OR MASS STORAGE FILE.

6. WHEN THE 'LABEL RECORDS ARE STANDARD' CLAUSE IS USED, THE CDC COMPILER ASSUMES A TAPE LABEL THAT HAS PREVIOUSLY BEEN DEFINED WITH A NOS/BE SYSTEM 'LABEL' CONTROL CARD. THE OPPOSITE IS TRUE IN IBM COBOL WHERE ALL DISK FILES MUST BE DESCRIBED AS 'LABEL RECORDS ARE STANDARD'.

7. THE 'BLOCK CONTAINS XXX RECORDS/CHARACTERS' CLAUSE MAY NOT HAVE THE SAME MEANING IN CDC COBOL AS ON THE 360. OMISSION OF THE BLOCK CONTAINS CLAUSE WILL RESULT IN A BLOCK SIZE BEST SUITED TO THE NOS/BE DEVICE TO WHICH THE FILE IS ASSIGNED.

FOR S AND L TAPES, THE BLOCK CONTAINS XXX RECORDS CLAUSE PROVIDES THE MOST EFFICIENT OPERATION.

8. THE FIRST ENTRY IN THE 'PROCEDURE DIVISION' MUST BE A PARAGRAPH NAME. IF NOT, A DIAGNOSTIC MESSAGE WILL BE GENERATED.

FOR EXAMPLE -

PROCEDURE DIVISION.

BEGIN-PROGRAM.

OPEN INPUT,

9. PRINTER CARRIAGE CONTROL IS ACHIEVED BY THE SPECIAL CHARACTERS THAT APPEAR IN THE FIRST POSITION OF THE RECORD TO BE PRINTED. CDC HAS THE FOLLOWING CONVENTIONS:

| | | |
|-------|---------------------|-----------------|
| 1 | PAGE EJECT | |
| BLANK | SPACING ONE LINE | |
| 0 | SPACING TWO LINES | |
| - | SPACING THREE LINES | (NOT FOR 200UT) |
| + | NO ADVANCE | |

SOME OTHER CARRIAGE CONTROL CHARACTERS ARE LESS FREQUENTLY USED.

WHEN USING THE CLAUSE 'WRITE XXXX AFTER ADVANCING X LINES', WHERE 'X' IS A NUMBER GREATER THAN ZERO, THE CARRIAGE WILL SPACE THE INDICATED NUMBER OF LINES. PAGE EJECTION HOWEVER REQUIRES A '1' IN THE FIRST POSITION OF THE OUTPUT LINE. THE IBM COBOL 'AFTER ADVANCING 0 LINES' MUST BE CHANGED. FOR EXAMPLE:

WRITE XXXX AFTER ADVANCING SKIP0.

WHERE SKIP0 IS DEFINED IN THE SPECIAL-NAMES PARAGRAPH.

SOURCE-COMPUTER. 6700.

OBJECT-COMPUTER. 6700.

SPECIAL-NAMES.

'1' IS SKIP0.

10. QUOTES MUST BE REMOVED FROM THE 'IMPLEMENTOR NAMES' OR 'EXTERNAL-FILE-NAMES' IN THE SELECT CLAUSES. QUOTES ARE NOT NEEDED ON ENTRY POINT NAMES USED IN CALL OR ENTER STATEMENTS. QUOTES ARE NOT NEEDED ON THE 'PROGRAM-ID', WHICH MAY BE UP TO 30 CHARACTERS LONG, ALTHOUGH ALL BUT THE FIRST 7 WILL BE IGNORED BY THE COMPILER.

11. INPUT AND OUTPUT PROCEDURES ARE SLOWER THAN THE 'USING' AND 'GIVING' OPTIONS, BUT THE INPUT AND OUTPUT PROCEDURES ALLOW SOME MANIPULATION OF DATA. IF INPUT PROCEDURE OR OUTPUT PROCEDURE IS SPECIFIED, THE USER MUST PROVIDE SUCH PROCEDURES IN THE FORM OF SECTIONS WITH SECTION HEADERS.

12. ALL VARIABLE SUBSCRIPTS SHOULD BE DEFINED AS 'COMPUTATIONAL-1' INDEPENDENT DATA ELEMENTS. IN ADDITION, VARIABLE SUBSCRIPTS SHOULD BE INCREMENTED BY SIMILARLY DEFINED ELEMENTS. FOR EXAMPLE, TO VARIABLE SUBSCRIPT 'A' BY ONE, USE:

77 A PICTURE S9, COMP-1, VALUE ??.

77 ONE PICTURE S9, COMP-1, VALUE 1.

ADD ONE TO A.

MOVE DATA-NAME (A) TO

IF VARIABLE SUBSCRIPTS ARE DEFINED AS NUMERIC DISPLAY ITEMS, THE COMPILER GENERATES OBJECT CODE TO CONVERT THE ITEMS TO COMPUTATIONAL-1, PERFORM THE SUBSCRIPTING OPERATION, AND THEN CONVERT BACK TO DISPLAY CODE. WHEN A FIXED SUBSCRIPT IS USED, IT MAY BE IN NUMERIC DISPLAY CODE SINCE THE COMPILER CONVERTS IT TO 'COMPUTATIONAL-1' AND IT IS NOT ALTERED BY PROGRAM EXECUTION.

13. ANY IBM S/360 PROGRAM USING INDEXED SEQUENTIAL (ISAM) WILL REQUIRE SOME CONVERSION TO USE CDC INDEXED SEQUENTIAL FILE ORGANIZATION. (SEE COBOL, CHAPTER II-2 AND RMC08, CHAPTER 7.)

14. COBOL DEBUGGING AIDS DO NOT INCLUDE THE SPECIAL VERBS 'TRACE', 'EXHIBIT', 'EXHIBIT NAMED', WHICH ARE IBM EXTENSIONS. SEE COBCL, CHAPTER II-14.

15. THE LFN PARAMETER ON THE FILE CARD ALLOWS FILE NAME SUBSTITUTION AT EXECUTE TIME. FOR EXAMPLE, A RELOCATABLE COBOL OBJECT PROGRAM WAS COMPILED WITH

SELECT TRANSFILE ASSIGN TO INPUT.

TO SUBSTITUTE A PREVIOUSLY CATALOGED DISK FILE FOR THE FILE 'INPUT' WITHOUT RECOMPILING, THE FOLLOWING CONTROL CARDS COULD BE USED:

ATTACH,BIN,COBOLBIN,ID=XXXX.
ATTACH,FIL,DISKTRANSFIL,ID=XXXX.
FILE,INPUT,LFN=FIL.
LDSET,FILES=INPUT.
BIN.

NOTE: THE FILE TO BE SUBSTITUTED MUST HAVE COMPATIBLE RECORD MANAGER BLOCK AND RECORD TYPES OR OTHER PARAMETERS WILL BE REQUIRED.

NOTE: DO NOT ATTEMPT TO CHANGE THE NAME OF AN EXISTING SIS OR DIRECT ACCESS FILE.

*** COBOL 4 VERSUS COBOL 3 ***

1. COBOL 4 HAS 54 ADDITIONAL RESERVED WORDS. SEE COBOL: APPENDIX C.
2. ALL COBOL 4 PROGRAMS WILL USE RECORD MANAGER FOR INPUT/OUTPUT FUNCTIONS.

COBOL 4 USES THE 'BLOCK CONTAINS' CLAUSE TO DETERMINE THE RECORD MANAGER BLOCK TYPE 'BT' AND THE 'RECORD CONTAINS' CLAUSE TO DETERMINE THE RECORD MANAGER RECORD TYPE 'RT' TO BE ASSIGNED.

WHEN PROCESSING FILES ON NOS/BE DEVICES WITH A COBOL 4 PROGRAM, THE 'BLOCK CONTAINS' CLAUSE SHOULD BE OMITTED OR STATED AS 'BLOCK CONTAINS NNN CHARACTERS' (WHERE NNN IS THE PRU SIZE OF THE NOS/BE DEVICE BEING USED) FOR EFFICIENT USE OF THE DEVICE.

WHEN PROCESSING FILES ON 'S' OR 'L' TAPES, THE 'BLOCK CONTAINS' CLAUSE SHOULD BE STATED AS 'BLOCK CONTAINS NNN RECORDS'.

FOR COBOL USE OF RECORD MANAGER, SEE RMC0B.

3. NOS/BE 1.0 UNIT RECORD DEVICE FILES MUST BE PROCESSED WITH A ZERO-BYTE TERMINATED RECORD TYPE DESIGNATION, 'RT=Z'. OTHER FILES MAY NEED TO BE DESCRIBED AS CONTAINING ZERO-BYTE TERMINATED RECORDS. ANY FILE WHICH ORIGINATED ON THE CARD READER BUT NO LONGER HAS THE LOGICAL FILE NAME 'INPUT' OR ANY FILE WHOSE ULTIMATE DESTINATION IS TO BE 'OUTPUT' OR 'PUNCH' SHOULD BE DESIGNATED AS CONTAINING ZERO-BYTE TERMINATED RECORDS.

COBOL 4 AUTOMATICALLY FORCES 'RT=Z' FOR ANY FILE ASSIGNED TO 'INPUT', 'OUTPUT', 'PUNCH', OR 'PUNCHB' IN THE 'SELECT' STATEMENT. SHOULD THE PROGRAMMER NEED TO DESIGNATE 'Z' TYPE RECORDS ON OTHER THAN THESE FILES HE MUST CHOOSE ONE OF THE FOLLOWING MEANS--

- A) A '-FZ' APPENDED TO THE LOCAL FILE NAME IN THE 'SELECT' CLAUSE FORCES 'Z' TYPE RECORDS AND OVERRIDES THE BLOCK CONTAINS CLAUSE.

EXAMPLE--SELECT COBOL3DISKFILE ASSIGN TO DISK1-FZ.

NOTE--THE LOCAL FILE NAME USED ON NOS/BE CONTROL CARDS DOES NOT USE THE '-FZ', I.E., THE ATTACH WOULD READ--

ATTACH,DISK1,ANYFILE,ID=XXXX.

- B) USE OF 'FILE' AND 'LDSET' CARDS FOR THE LOCAL FILE NAME OF THE FILE WITH 'Z' TYPE RECORDS WILL OVERRIDE THE COBOL GENERATED BLOCK AND RECORD TYPES.

EXAMPLE--

```
JOB CARD
CHARGE CARD
COBOL(LR,U)
ATTACH(DISK1,ANYFILE,ID=XXXX)
FILE(DISK1,BT=C,RT=Z)
LDSET(FILES=DISK1)
LGO.
' 7/8/9      EOR
  (COBOL SOURCE PROGRAM)
" 6/7/8/9    EOF
```

4. COBOL 3 IGNORED THE BLOCK CONTAINS CLAUSE AND PROCESSED SCOPE CODED MASS STORAGE OR TAPE FILES AS ZERO BYTE TERMINATED RECORD FILES. PROCESSING OF THESE FILES MAY BE DONE AS DESCRIBED ABOVE OR BY USING THE 'Z' PARAMETER ON THE COBOL CONTROL CARD.
5. IN COBOL 3 IF A NONEXISTENT SEQUENTIAL FILE WAS OPENED AS 'INPUT' THE 'AT END' BRANCH WAS TAKEN ON THE FIRST READ FOR THE FILE, REGARDLESS OF THE SPECIFICATION OF 'OPTIONAL' IN THE 'SELECT' PARAGRAPH. IN COBOL 4 IT IS FATAL TO OPEN A FILE AS 'INPUT' IF THE FILE DOES NOT EXIST. IF 'OPTIONAL' IS SPECIFIED, THE 'OPEN' IS LEGAL AND THE 'AT END' IS TAKEN ON THE FIRST READ.
6. THE VALUE SPECIFIED IN THE 'ADVANCING' PHRASE OF THE 'WRITE' VERB NOW CAUSES THE EXACT NUMBER OF LINES TO BE ADVANCED ON THE PRINTER. SINCE COBOL 3 COUNTED BLANK LINES HERE, ANY PAPER ADVANCEMENT IN COBOL 4 IS REDUCED BY ONE FROM COBOL 3. THE COMBINATION OF A WRITE LINEX AFTER ADVANCING NNN LINES FOLLOWED BY A WRITE LINEXX BEFORE ADVANCING NNN LINES WILL RESULT IN LINEXX BEING WRITTEN OVER THE TOP OF LINEX.
NOTE--DO NOT MAKE THE AFTER ADVANCING CHANGES TO YOUR PROGRAM IF YOU ARE USING THE Z PARAMETER ON THE COBOL CARD.
7. UNDER SCOPE 3.3, THE SYSTEM DEFAULT COLLATING SEQUENCE WAS THE CDC STANDARD COBOL COLLATING SEQUENCE (ALPHABETICS PRECEDE NUMERICS). UNDER NOS/BE 1.0, THE SYSTEM DEFAULT COLLATING SEQUENCE IS THE ASCII6 COLLATING SEQUENCE (NUMERICS PRECEDE ALPHABETICS). TO OVERRIDE THE DEFAULT COLLATING SEQUENCE AND INSURE THAT A PROGRAM UTILIZES THE CDC STANDARD COBOL COLLATING SEQUENCE INCLUDE A 'U' PARAMETER ON THE COBOL CONTROL CARD.

```
EXAMPLE---COBOL(U)
          COBOL(I=LFN,LR,U)
```


8. ALL 7/8/9 CARDS ON THE INPUT FILE ARE CONSIDERED TO BE END-OF-FILE MARKERS BY COBOL VERSION 4. PREVIOUS VERSIONS WOULD CONSIDER ONLY A 7/8/9 WITH A LEVEL 15 AS AN END-OF-FILE.
9. SCOPE 3.3 ALLOWED THE COBOL PROGRAMMER TO READ FROM FILE INPUT AND WRITE TO FILE OUTPUT WITHOUT REOPENING THEM AFTER EXECUTION OF A COBOL CLOSE STATEMENT. TO USE A FILE THAT HAS BEEN CLOSED UNDER COBOL VERSION 4 ANOTHER OPEN STATEMENT IS REQUIRED.
10. SCOPE 3.3 VERIFIED ALL SUBSCRIPT VALUES TO BE IN THE RANGE OF THE WORKING STORAGE AREA. WHEN A SUBSCRIPT WAS OUT OF RANGE, THE DIAGNOSTIC NEGATIVE, ZERO, OR HUGE SUBSCRIPT WAS ISSUED AND THE SOURCE LINE CAUSING THE ERROR WAS INDICATED THEREBY TERMINATING THE RUN. UNDER NOS/BE 1.0 NO SUBSCRIPT VALUE CHECKING IS DONE UNLESS DB IS SELECTED ON THE COBOL CONTROL CARD. WHEN DB IS NOT SELECTED, THE PROGRAM EXECUTES FASTER BECAUSE LESS CODE HAS BEEN GENERATED. HOWEVER, NO DB CAN LEAD TO UNDETECTED ERRORS IF YOU HAVE RELIED ON THE SUBSCRIPT DIAGNOSTIC MESSAGE IN THE PAST. A GOOD IDEA MIGHT BE TO RUN WITH DB SELECTED DURING TEST PHASE AND NOT SELECTED FOR THE PRODUCTION RUNS.
11. PREVIOUS VERSIONS OF COBOL MADE NO ATTEMPT AT INITIALIZING WORKING-STORAGE AREAS. COBOL VERSION 4 NOW INITIALIZES WORKING-STORAGE AREAS WHICH DO NOT HAVE VALUE CLAUSES TO BLANKS. IF A PROGRAM EXPECTS AN UNINITIALIZED AREA TO CONTAIN OTHER THAN SPACES THE PROGRAM MUST EXPLICITLY CAUSE THAT AREA TO CONTAIN THE CORRECT VALUE. THIS WILL AFFECT THOSE USERS WHO WERE DEPENDENT UPON SETCORE TO INITIALIZE CORE BEFORE PROGRAM EXECUTION.
12. UNDER SCOPE 3.3 WHEN A FIELD BEING MOVED TO A NUMERICALLY EDITED FIELD CONTAINED A NON-NUMERIC CHARACTER THE MESSAGE 'NON-NUMERIC IN FIELD TO BE EDITED' WAS ISSUED AND THE OPERATOR GIVEN THE OPTION TO GO OR DROP THE JOB. UNDER NOS/BE 1.0 THE SAME MESSAGE IS ISSUED FOR THIS CONDITION AND THE RUN TERMINATED. IF FOR SOME REASON THE RUN MUST BE ALLOWED TO CONTINUE WITH THE NON-NUMERIC ERROR, AN 'EE=NN' PARAMETER MAY BE SPECIFIED IN THE EXECUTION CALL TO ALLOW A MAXIMUM NUMBER OF NON-FATAL EXECUTION ERRORS. WHEN THE COUNT SPECIFIED BY THE INTEGER IS EXCEEDED AND AN ERROR OCCURS, THE RUN IS TERMINATED.

EXAMPLE--LG0(EE=10)

13. IN COBOL 4, THE DECLARATIONS 'RECORD CONTAINS DEPENDING UPON DATA-NAME' AND 'OCCURS DEPENDING UPON DATA-NAME' ARE NOT ACCEPTABLE IF THE DATA-NAME IS COMP-1 OR COMP-2. THE DATA-NAME MUST BE NUMERIC AND MUST BE 6 OR FEWER CHARACTERS IN LENGTH.

*** SAMPLE COBOL DECK SETUPS ***

COMPILE, NO LIST, AND EXECUTE

| | |
|---------------------------|--------------------------------------|
| XXXXSTS,CM100000,T100,P2. | NAME / CODE |
| CHARGE,XXXX,JJJJJJJJJJ. | |
| COBOL(L=0) | ** L=0 SUPPRESSES SOURCE LISTING |
| LGO. | ** LOAD AND EXECUTE FROM FILE 'LGO'. |
| * 7/8/9 EOR | |
| (COBOL SOURCE DECK) | |
| * 7/8/9 EOR | |
| (DATA DECK AS NEEDED) | |
| " 6/7/8/9 EOF | |

COMPILE, PUNCH BINARY DECK AND EXECUTE USING A DATA TAPE

| | |
|---|--|
| XXXXST2,CM100000,T100,MT1. | NAME / CODE |
| CHARGE,XXXX,JJJJJJJJJJ. | |
| COBOL(R=PUNCHB) | ** SEE COBOL CARD OPTIONS ABOVE |
| LABEL,DATIN,R,D=HY,L=XXXXDATIN,VSN=CB7777,NORING. | |
| PUNCHB. | ** LOAD AND EXECUTE FROM FILE 'PUNCHB' |
| * 7/8/9 EOR | |
| (COBOL SOURCE DECK) | |
| * 7/8/9 EOR | |
| (DATA DECK AS NEEDED) | |
| " 6/7/8/9 EOF | |

COMPILE AND EXECUTE WITH SUBPROGRAMS

| | |
|----------------------------------|---|
| XXXXST3,CM61000,T200,P2. | NAME / CODE |
| CHARGE,XXXX,JJJJJJJJJJ. | |
| COBOL. | ** COMPILE MAIN PROGRAM |
| COBOL(SUB) | ** COMPILE COBOL SUBPROGRAM |
| COBOL,SUB. | |
| LGO. | ** LOAD AND EXECUTE ENTIRE OBJECT PROGRAM |
| * 7/8/9 EOR | |
| (COBOL MAIN PROGRAM SOURCE DECK) | |
| * 7/8/9 EOR | |
| (COBOL FIRST SUBPROGRAM) | |
| * 7/8/9 EOR | |
| (COBOL SECOND SUBPROGRAM) | |
| * 7/8/9 EOR | |
| (DATA DECK AS NEEDED) | |
| " 6/7/8/9 EOF | |

COMPILE, CATALOG OBJECT CODE, EXECUTE

```
XXXXST1,CM100000,T100.                NAME / CODE
CHARGE,XXXX,JJJJJJJJJJ.
REQUEST,BINLIB,*PF.                    ** GET SPACE FOR PERMANENT FILE
COBOL(P=BINLIB)                        ** COBOL OUTPUT TO FILE
CATALOG(BINLIB,DAYACCOUNT,ID=XXXX,XR=RDONLY,PW=RDONLY)
BINLIB.                                ** LOAD AND EXECUTE FILE
*   7/8/9      EOR
   (COBOL SOURCE DECK)
*   7/8/9      EOR
   (DATA, AS NEEDED)
"   6/7/8/9    EOF
```

EXECUTE CATALOGED OBJECT CODE

```
XXXXCB4,CM100000,P2.                NAME / CODE
CHARGE,XXXX,JJJJJJJJJJ.
ATTACH,DAYAC,DAYACCOUNT,ID=XXXX.
DAYAC.
*   7/8/9      EOR
   (DATA, AS NEEDED)
"   6/7/8/9    EOF
```

*** OVERLAY FOR COBOL ***

THE 'SEGMENTATION' FEATURE OF COBOL IS A MEANS OF OVERLAYING PROCEDURE DIVISION CODE. WHILE THIS IS EASY TO ACCOMPLISH, THE MAJOR DISADVANTAGE IS THAT NON-PROCEDURE DIVISION PORTIONS OF SUBROUTINES CANNOT BE OVERLAYED. HENCE THE (0,0) LEVEL CONTAINS THE DATA DIVISION OF THE MAIN AND ALL SUBROUTINES. TO MORE EFFICIENTLY UTILIZE CORE STORAGE, FORTRAN OVERLAY TECHNIQUES ALLOW OVERLAY OF WHOLE SUBPROGRAMS.

GIVEN A MAIN PROGRAM AND TWO SUBPROGRAMS IN COBOL, AN EFFECTIVE OVERLAY IS CREATED BY ADDING TO THE MAIN LINK A FORTRAN SUBROUTINE TO CALL THE OVERLAYS, AND ADDING TO EACH OVERLAY LEVEL A DUMMY MAIN FORTRAN PROGRAM. COBOL CANNOT BE USED FOR THE DUMMY MAIN PROGRAM IN A LEVEL. AS LONG AS THE FORTRAN SEGMENTS OF THE PROGRAM DO NO FILE MANIPULATION OR ARITHMETIC OPERATIONS, NO COMMON OR FILE EQUIVALENCE IS REQUIRED.

COMPILE WITH SUBROUTINES AND OVERLAY

| | NAME / CODE |
|-------------------------|--|
| USERTRY,CM61000,T200. | |
| CHARGE,USER,JJJJJJJJJJ. | |
| COPYR(INPUT,LGO,C) | ** INSERT FIRST OVERLAY CARD |
| COBCL. | ** COMPILE COBOL MAIN PROGRAM |
| FTN. | ** COMPILE FORTRAN SUBROUTINE |
| FTN. | ** COMPILE DUMMY MAIN PROGRAM WITH OVERLAY |
| COBCL(SUB) | ** COMPILE FIRST SUBROUTINE |
| FTN. | ** COMPILE DUMMY MAIN PROGRAM WITH OVERLAY |
| COBCL(SUB) | ** COMPILE SECOND SUBROUTINE |
| LGO. | ** LOAD AND EXECUTE |
| * 7/8/9 EOR | |
| OVERLAY(DON,0,0) | |
| * 7/8/9 EOR | |

COBOL SOURCE DECK WHICH CALLS FORT

```

* 7/8/9 EOR
  SUBROUTINE FORT
C   FORTRAN PROGRAM MUST NOT DO ANY I/O, MERELY CALL OVERLAYS.
    CALL OVERLAY (3HDON,1,0)
    CALL OVERLAY (3HDON,2,0)
    RETURN
  END
* 7/8/9 EOR
  CVERLAY(DON,1,0)
  PROGRAM FORT1
  CALL SUB1
C   FORTRAN MAIN MUST DO NO I/O OR ARITHMETIC
  END
* 7/8/9 EOR

```

COBOL SUBPROGRAM 1

```

* 7/8/9 EOR
  OVERLAY(DON,2,0)
  PROGRAM FORT2
  CALL SUB2
  END
* 7/8/9 EOR

```

COBOL SUBPROGRAM 2

```

* 7/8/9 EOR

  (DATA, AS NEEDED)

" 6/7/8/9 EOF

```


*** COBOL 5 ***

COBOL5, WHICH IS CDC'S IMPLEMENTATION OF THE 1974 ANSI COBOL STANDARD, IS AVAILABLE AS A PERMANENT FILE ON ALL THREE 6000 COMPUTERS. COBOL5 IS IN ADDITION TO, BUT DOES NOT REPLACE, THE CURRENT SYSTEM COBOL. ALMOST ALL COBOL 4 PROGRAMS WILL NEED SOME CONVERSION TO COMPILE IN COBOL 5. COMPILATION IN COBOL 5 IS CONSIDERABLY SLOWER. EXECUTION MAY BE MUCH FASTER, PARTICULARLY WHERE SORTING IS INVOLVED.

CONTROL CARDS FOR NOS/BE ARE:

```
... AT LEAST CM65000 ON JOB CARD
...
ATTACH,CBL,434COBOL5,ID=CSYS.
LIBRARY,CBL.
COBOL5,<OPTIONS>.
LGO.
' 7/8/9 EOR
(COBOL 5 SOURCE PROGRAM)
" 6/7/8/9 EOF
```

CAUTION! WHEN USING COBOL5 FROM A USER LIBRARY (AS ABOVE), IT IS NECESSARY TO ATTACH THE LIBRARY EVEN WHEN EXECUTING FROM AN ABSOLUTE COBOL5 OBJECT FILE. USE THE SAME LFN FOR THE LIBRARY WHEN COMPILING AND EXECUTING, E.G.,

```
ATTACH,MYPROG,ID=XXXX.      ** GET ABSOLUTIZED PROGRAM
ATTACH,CBL,434COBOL5,ID=CSYS. ** GET COBOL5 LIBRARY
MYPROG.                    ** EXECUTE PROGRAM
```

CONTROL CARD PARAMETERS

THE PARAMETERS AVAILABLE ON THE COBOL5 CONTROL CARD DO NOT ALL CORRESPOND TO THOSE USED BY COBOL 4. IN THE DISCUSSION BELOW, SOME OF THE PARAMETERS HAVE TWO DEFAULTS. THEY WILL BE SHOWN AS (DEFAULT: DEFAULT1/DEFAULT2) WHERE DEFAULT1 IS WHEN THE KEYWORD IS OMITTED AND DEFAULT2 IS WHEN THE KEYWORD IS GIVEN ALONE. FOR EXAMPLE, THE DEFAULTS FOR THE 'I' OPTION ARE INPUT/COMPILE. THIS MEANS THAT WHEN THE 'I' PARAMETER IS OMITTED (COBOL5,....), THE DEFAULT IS 'INPUT' AND WHEN THE 'I' PARAMETER IS GIVEN ALONE (COBOL5,I,....), THE DEFAULT IS 'COMPILE'. SOME OF THE MORE FREQUENTLY USED PARAMETERS ARE:

| _OPTION_ | -----ACTION----- |
|----------|---|
| B=LFN | FILE TO CONTAIN RELOCATABLE BINARY OBJECT PROGRAM (DEFAULTS: LGO/BIN) USE B=0 TO SUPPRESS BINARY OUTPUT |
| I=LFN | FILE CONTAINING SOURCE PROGRAM (DEFAULTS: INPUT/COMPILE) |
| L=LFN | FILE TO CONTAIN COMPILATION LISTINGS (DEFAULTS: OUTPUT/LIST) USE L=0 TO SUPPRESS THE LISTING |
| LO= | LISTING OPTIONS M - STORAGE MAP O - OBJECT CODE LISTING (USE ONLY IF REQUESTED BY CODE 1892) R - CROSS-REFERENCE LIST S - LIST SOURCE PROGRAM WHEN MORE THAN ONE IS SPECIFIED, SEPARATE THEM WITH /, E.G., LO=M/S. (DEFAULTS: S / 'M/R/S') |
| PD= | PRINT DENSITY. 3, 4, 6, OR 8 LINES/INCH MAY BE SELECTED. (DEFAULTS: 6/8) |
| SY | SYNTAX CHECK ONLY. DO NOT PRODUCE OBJECT CODE. CUTS COMPILATION TIME ROUGHLY IN HALF. |

SEE COBOL5 REFERENCE MANUAL, CHAPTER 12, FOR ADDITIONAL OPTIONS.

CONTROL CARD EXAMPLES

COBOL5. DEFAULTS TO: I=INPUT,L=OUTPUT,LO=S,B=LGO
SOURCE PROGRAM FROM FILE 'INPUT'
COMPILATION LISTING ON FILE 'OUTPUT'
SOURCE LISTING SELECTED
RELOCATABLE BINARY OBJECT CODE ON FILE 'LGO'

COBOL5(I=INP,L=OUTP,LO=M/O/R/S,SY,PD=8)
SOURCE PROGRAM IS ON FILE 'INP'
COMPILATION LISTING ON FILE 'OUTP'
LIST OPTIONS M, O, R, S SELECTED
SYNTAX CHECK ONLY (NO BINARY OUTPUT)
PRINT DENSITY IS 8 LINES PER INCH

COBOL 5 VERSUS COBOL 4

COBOL 4 AND COBOL 5 HAVE SUBSTANTIAL DIFFERENCES BECAUSE THEY WERE DESIGNED TO COMPLY WITH DIFFERENT VERSIONS OF THE ANSI COBOL STANDARD. A GOOD CONDENSED LIST OF THE MAJOR DIFFERENCES IS GIVEN IN CHAPTER 3 IN THE COBOL4 TO COBOL5 CONVERSION AIDS REFERENCE MANUAL. AMONG THESE DIFFERENCES ARE:

DUPLICATE KEYS ARE NO LONGER ALLOWED FOR INDEXED SEQUENTIAL FILES.

USER-DEFINED LABELS ARE NO LONGER ALLOWED.

SKIP VERB HAS BEEN DELETED.

EXAMINE VERB HAS BEEN REPLACED BY INSPECT.

STRING, UNSTRING, MERGE, AND INITIALIZE HAVE BEEN ADDED.

COLLATING SEQUENCE CONTROL HAS BEEN EXPANDED AND MAY BE DYNAMICALLY CONTROLLED.

CHARACTER CODE CONVERSION CAN BE SPECIFIED FOR A TAPE IN UNIVAC FIELDATA FORMAT.

REFERENCES: COBOL 5 REFERENCE MANUAL CDC 60497100
COBOL 4 TO COBOL 5 CONVERSION AID CDC 19265021A
FEDERAL INFORMATION PROCESSING STANDARDS, DEC 1975,
PUBLICATION NUMBER 21-1 (GIVES INFORMATION REGARDING
FEDERAL GUIDELINES FOR USING ANSI 74 COBOL COMPILER)

*** COBOL4 TO COBOL5 CONVERSION AID ***

A CONVERSION AID PROGRAM IS AVAILABLE TO ASSIST IN CONVERTING PROGRAMS FROM COBOL4 TO COBOL5. THE DETAILED DESCRIPTION OF ITS OPERATION IS FOUND IN 'COBOL VERSION 4 TO COBOL VERSION 5 CONVERSION AID' REFERENCE MANUAL (CDC PUBLICATION NO. 19265021).

FIELD LENGTH REQUIRED: 77000 CM WORDS.

TAPE11 IS THE DEFAULT INPUT SOURCE TO THE TRANSLATOR.

TAPE25 IS THE DEFAULT CONVERTED OUTPUT FILE.

COBOL COPY LIBRARIES MUST BE CONVERTED VIA A SPECIAL ROUTINE (SEE CBLCOP CONTROL CARD IN CONVERSION AID REFERENCE MANUAL).

EXAMPLE 1. CONVERT SOURCE DECK ON CARDS AND CALL COBOL5 FOR COMPILE.

| | NAME / CODE |
|---------------------------------------|------------------------------------|
| JORNAME,CM77000,..... | |
| CHARGE,..... | |
| COPYCR,INPUT,TAPE11. | ** COPY INPUT CARDS TO FILE TAPE11 |
| REWIND,TAPE11. | |
| ATTACH,LCS,ID=CSYS. | ** ATTACH CONVERSION |
| ATTACH,TAPE10,LCSYNTAX,ID=CSYS. | ** PROGRAM FILES AND |
| LCS. | ** EXECUTE (INPUT ON TAPE11) |
| REWIND,TAPE25. | |
| ATTACH,CBL,434COBOL5,ID=CSYS. | ** ATTACH COBOL5 |
| LIBRARY,CBL. | ** LIBRARY AND |
| COBOL5(I=TAPE25) | ** CALL COBOL5 |
| * 7/8/9 EOR | |
| COBOL4 SOURCE DECK | |
| * 7/8/9 EOR | |
| UPD,CBLLIST (LISTS ALL COBOL4 SOURCE) | |
| CBL | |
| " 6/7/8/9 EOF | |

EXAMPLE 2. PROGRAM TO BE CONVERTED IS ON AN UPDATE PROGRAM LIBRARY. TRANSLATE AND COMPILE.

| | NAME / CODE |
|---------------------------------|--|
| JORNAME,CM77000,..... | |
| CHARGE,..... | |
| ATTACH,OLDPL,..... | |
| UPDATE(Q) | ** SELECT A DECK TO BE TRANSLATED |
| ATTACH,LCS,ID=CSYS. | |
| ATTACH,TAPE10,LCSYNTAX,ID=CSYS. | |
| LCS,COMPILE. | ** CONVERT SELECTED PROGRAM (FROM COMPILE) |
| REWIND,TAPE25. | |
| ATTACH,CBL,434COBOL5,ID=CSYS. | |
| LIBRARY,CBL. | |
| COBOL5(I=TAPE25) | |
| * 7/8/9 EOR | |
| *COMPILE DECKNAME | |
| * 7/8/9 EOR | |
| UPD,CBLLIST | |
| CBL | |
| " 6/7/8/9 EOF | |

EXAMPLE 3. CONVERT SOURCE PROGRAM FROM UPDATE PL AND CREATE NEW PL.

| | NAME / CODE |
|---------------------------------|----------------------------------|
| JOBNAME,CM77000,..... | |
| CHARGE,..... | |
| REQUEST,NEWPL,*PF. | |
| ATTACH,OLDPL,..... | |
| UPDATE(IQ,S=TAPE11,C=0) | ** SELECT DECK (+ *DECK CARD) TO |
| UNLOAD,OLDPL. | BE CONVERTED |
| REWIND,TAPE11. | |
| ATTACH,LCS,ID=CSYS. | |
| ATTACH,TAPE10,LCSYNTAX,ID=CSYS. | |
| LCS. | |
| REWIND,TAPE25. | |
| UPDATE(I=TAPE25,N) | ** SEE NOTE BELOW |
| CATALOG,NEWPL,..... | |
| * 7/8/9 EOR | |
| *COMPILE DECKNAME | |
| * 7/8/9 EOR | |
| UPD,CBLLIST | |
| CBL | |
| " 6/7/8/9 EOF | |

NOTE: UPDATE WILL CREATE NEW PL OF TRANSLATED DECK. THIS WILL BE THE ONLY SOURCE DECK FROM THE OLDPL TO APPEAR IN THE NEWPL.

***** UTILITIES *****

A WIDE VARIETY OF UTILITY PROGRAMS IS AVAILABLE AS A PART OF THE SYSTEM OR IN LIBRARIES OR OTHER FILES. THIS CHAPTER DESCRIBES MANY OF THE SYSTEM UTILITIES (NO ATTACH REQUIRED). SEE CHAPTER 10, CCLIB AND CCLIB/U FOR ADDITIONAL UTILITIES.

UTILITY OPERATIONS CAN BE PERFORMED WITH NAMED FILES, EACH ASSIGNED TO A SPECIFIC PERIPHERAL DEVICE (MAY NEED REQUEST CARDS WHICH MUST PRECEDE THE UTILITY CONTROL CARDS). MOST COPY UTILITIES USE A FIELD LENGTH OF 20000. ALL NOS/BE UTILITIES DESCRIBED BELOW ARE FOR SEQUENTIAL FILES ONLY. SIS FILES ARE MANIPULATED BY COPYBE OR FORM. FOR RANDOM FILES, SEE FORM; SECTORAN DIRECTIVE OF EDITLIB (CCRM, 7-11); COPYE (CCRM, 6-4); COPYS (CCRM, 10-21); COPYBER (CCRM, 10-21). COPYRM (CCRM, 6-6) MAY BE USED TO COPY MOST FILES.

*** NOS/BE UTILITIES ***

SEE NOSBE, CHAPTER 4, AND SCOPE, CHAPTER 8, FOR A FULL DESCRIPTION OF THE FOLLOWING UTILITIES.

COPYCF(FILIN,FILOUT,N)
COPYBF(FILIN,FILOUT,N)

COPY CODED FILES
COPY BINARY FILES

COPYCR(FILIN,FILOUT,N)
COPYER(FILIN,FILOUT,N)

COPY CODED RECORDS
COPY BINARY RECORDS

FILIN - INPUT FILE (DEFAULT: INPUT)
FILOUT - OUTPUT FILE (DEFAULT: OUTPUT)
N - NUMBER OF FILES OR RECORDS (DEFAULT: 1)

SKIFF(FILE,N,LEVEL,TYPE)
SKIPB(FILE,N,LEVEL,TYPE)

SKIP FORWARD
SKIP BACKWARD

FILE - NAME OF FILE TO BE POSITIONED
N - NUMBER OF RECORDS/FILES TO BE SKIPPED
LEVEL - END-OF-RECORD LEVEL NUMBER
0 OR OMITTED - NORMAL END-OF-RECORD
17 - END-OF-FILE
TYPE - ONE OF
C CODED FILES (7-TRACK, EVEN PARITY)
B BINARY FILES (7-TRACK, ODD PARITY)

COPYL(OLDLIB,BINARY,NEWLIB) SELECTIVE REPLACEMENT OF ROUTINES
 COPYL(OLDLIB,BINARY,NEWLIB,LAST,FLAG)

COPYLM(OLDLIB,BINARY,NEWLIB)
 COPYLM(OLDLIB,BINARY,NEWLIB,LAST,FLAG)

OLDLIB - OLD MASTER BINARY (DEFAULT: OLD)
 BINARY - REPLACEMENT FILE (DEFAULT: LGO)
 NEWLIB - CREATE UPDATED MASTER FILE (DEFAULT: NEW)
 LAST - NAME OF LAST RECORD ON OLDLIB TO BE PROCESSED.
 (DEFAULT: ALL RECORDS ON OLDLIB ARE PROCESSED)
 FLAG - PROCESSING OPTIONS (DEFAULT: OPTION NOT SELECTED)
 R REWIND OLDLIB AND NEWLIB BEFORE PROCESSING
 (BINARY IS ALWAYS REWOUND BEFORE AND AFTER)
 A APPEND TO END OF NEWLIB ALL BINARY RECORDS THAT DID
 NOT MATCH ANY OF OLDLIB.
 TO USE BOTH R AND A OPTIONS COMBINE THE LETTERS "RA"

COPYL PROCESSES THE MASTER FILE (OLDLIB) FORWARD ONLY, BUT WILL
 SEARCH BINARY AND REPLACE ALL ROUTINES OF THE SAME NAME FROM THEIR
 FIRST OCCURRENCE ON OLDLIB.

COPYLM DIFFERS ONLY IN ITS HANDLING OF MULTIPLE OCCURRENCES OF A
 RECORD ON THE OLD MASTER. USING COPYLM, ALL OCCURRENCES OF THE
 ROUTINE ON OLDLIB ARE REPLACED BY THE FIRST MATCHING RECORD ON
 BINARY. THIS FACILITATES REPLACING A SUBROUTINE WHICH OCCURS IN
 SEVERAL OVERLAYS.

ITEMIZE(LFN) LIST CONTENTS OF BINARY FILE
 ITEMIZE(LFN,<PARAMETERS>)

OUTPUT INCLUDES RECORD NUMBER, NAME, LENGTH, PREFIX TABLE FOR
 RELOCATABLE BINARY OR USER LIBRARY. FOR SEQUENTIAL PL, ONLY DECK
 NAMES ARE LISTED. (SEE CORM, 7-10: CONTENT;
 CCLIB/P: LISTRIN;
 CORM, 3-2)

OPTIONAL PARAMETERS INCLUDE:

- E - LIST ENTRY POINTS FOR RELOCATABLE PROGRAMS OR IDENTs FOR
 SEQUENTIAL PL
- L=LFN - IF OTHER THAN "OUTPUT"
- N - ITEMIZE TO END-OF-INFORMATION
- N=NN - ITEMIZE NN FILES (DEFAULT: 1)

COPYN(F1,FILOUT,FILIN1,....,FILINN) COPY, MERGE OR SELECT LOGICAL
RECORDS FROM UP TO 10 BINARY
INPUT FILES.

THERE ARE NO ARGUMENT DEFAULTS.
F1 IS RECORD FORMAT; USE 0 (ZERO).
FILOUT THE OUTPUT FILE PRECEDES ALL INPUT FILES.
THE TEXT LOGICAL RECORD WHICH CONTAINS RECORD IDENTIFICATION
CARDS WILL BE FILE 'INPUT'. COMMAND SUMMARY WILL BE IN 'OUTPUT'.

THE RECORD IDENTIFICATION CARDS BEGIN IN COLUMN 1 AND HAVE THE
FOLLOWING FORMAT: P1,P2,P3
WHERE P1 IS RECORD NAME OR NUMBER, RELATIVE TO CURRENT POSITION,
TO START COPY
P2 IS LAST RECORD NAME OF A SET OR TOTAL NUMBER OF RECORDS
TO BE COPIED. IF BLANK, COPY ONLY P1
IF "***", COPY THRU END OF FILE.
P3 IS NAME OF INPUT FILE TO SEARCH (MAY BE OMITTED)
RECORD IDENTIFICATION CARDS MUST BE BLANK (THRU COLUMN 80)
FOLLOWING THE LAST SPECIFIED PARAMETER.

SEVERAL FILE MANIPULATION COMMANDS ARE ALSO ALLOWED.

SEARCH IN ORDER OF ROUTINES ON FILE WHEN POSSIBLE. COPYN WILL
END AROUND SEARCH, SO IS COSTLY IF MANY REWINDS ARE REQUIRED.

| | |
|------------------------|--|
| BKSP,LFN. | POSITION READY TO READ LAST PREVIOUS LOGICAL RECORD. IF FILE WAS AFTER EOF, POSITION JUST BEFORE THE EOF. |
| COMBINE,FILE1,FILE2,N. | REMOVE LOGICAL RECORD BOUNDARIES FROM FIRST N RECORDS ON FILE1, MAKING ONE LOGICAL RECORD ON FILE2 (BINARY). |
| COMPARE,FILE1,FILE2. | VERIFY THAT A COPY IS ERROR FREE. COMPARES RECORDS FROM CURRENT POSITION WITH ADDITIONAL OPTIONAL PARAMETERS TO CONTROL NUMBER OF RECORDS AND NO OF ERRORS. NOTE: DISK FILES SHOULD NOT BE COMPARED WITH TAPE FILES USING THIS UTILITY BECAUSE ERRONEOUS 'BAD COMPARE'S CAN OCCUR. |
| COPY,FILE1,FILE2. | COPY TO DOUBLE END-OF-FILE (OR END-OF- INFORMATION, IF ENCOUNTERED FIRST). (FOR A BETTER ROUTINE, SEE CCRM, 6-4: COPYE) |

*** OTHER UTILITIES ***

THE FOLLOWING UTILITIES WERE OBTAINED FROM THE UNIVERSITY OF WASHINGTON AND HAVE BEEN MADE A PART OF THE DTNSRDC 6000 COMPUTER OPERATING SYSTEMS (NO ATTACH REQUIRED): COPYE, COPYF, COPYR, COPYRM, COPYSE/COPYSBF/COPYSR (WHICH REPLACES AND EXPANDS THE CDC VERSION OF COPYSBF). FULL DOCUMENTATION MAY BE OBTAINED BY USING THE FOLLOWING CONTROL CARD:

BEGIN,UTILITY,,PROGDOC,OTHER,,*,OUTPUT.

WHERE * IS ONE OF: COPYE, COPYF, COPYR, COPYRM, COPYSE.

TO PRINT SEVERAL DOCUMENTS, USE:

BEGIN,UTILITY,,MANYDOC,OTHER,,INPUT,OUTPUT.

WHERE FILE 'INPUT' CONTAINS THE DESIRED DOCUMENT NAMES, ONE PER CARD.

COPYE(FILIN,FILOUT)

MAKE AN EXACT COPY FROM CURRENT POSITION
TO END-OF-INFORMATION

THERE ARE NO DEFAULT FILE NAMES.

IF FILIN IS RANDOM AND IF FILIN AND FILOUT ARE DISK AND ARE REWOUND, THEN FILOUT IS RANDOM.

MODE SWITCHING WILL OCCUR IF A PARITY ERROR OCCURS IN THE FIRST PRU OF FILIN.

FILOUT IS BINARY.

THIS IS THE EASIEST WAY TO PRODUCE AN EXACT COPY OF AN ARBITRARY NOS/BE FILE.

THIS IS THE FASTEST WAY TO COPY DISK-TO-DISK.

| | |
|-------------------------------|----------------------|
| COPYF(FILIN,FILOUT,N) | COPY N FILES |
| COPYF(FILIN,FILOUT,N,C) | COPY N CODED FILES |
| COPYR(FILIN,FILOUT,N,LEVEL) | COPY N RECORDS |
| COPYR(FILIN,FILOUT,N,LEVEL,C) | COPY N CODED RECORDS |
| COPYF(,FILOUT,N) | WRITE N EOF'S |
| COPYF(,FILOUT,N,C) | WRITE N CODED EOF'S |
| COPYR(,FILOUT,N,LEVEL) | WRITE N EOR'S |
| COPYR(,FILOUT,N,LEVEL,C) | WRITE N CODED EOR'S |

FILIN - INPUT FILE (NO DEFAULT)
 FILOUT - OUTPUT FILE (NO DEFAULT)
 N - NUMBER OF FILES/RECORDS TO COPY OR
 NUMBER OF EOF/EOR TO WRITE
 (DEFAULT: 1)
 C - COPY CODED FILES/RECORDS OR
 WRITE CODED EOF/EOR
 LEVEL - EOR LEVEL NUMBER (OCTAL)
 (DEFAULT: 0)
 (COPYR ONLY. IF USED, N MUST ALSO BE USED.)

THERE ARE NO DEFAULT FILE NAMES.
 NOS/BE FILES ONLY. NO S OR L TAPES.
 FOR COPYF, IF EOI IS ENCOUNTERED WITHOUT EOF JUST BEFORE IT, AN EOF
 IS ADDED TO FILOUT.
 FOR COPYR, IF LAST RECORD WRITTEN WAS OF A LEVEL LESS THAN THE
 SPECIFIED LEVEL, A ZERO-LENGTH RECORD OF THE SPECIFIED LEVEL IS
 WRITTEN.
 FOR COPYR, A RECORD LEVEL MAY NOT BE SPECIFIED WITHOUT A RECORD
 COUNT.

MESSAGES

| | | |
|-------------------------|-------|--|
| BCD MODE FILE M | COPYR | TAPE MODE SWITCHED TO CODED |
| | COPYF | |
| BIN MODE FILE M | COPYR | TAPE MODE SWITCHED TO BINARY |
| | COPYF | |
| EOF ENCOUNTERED | COPYR | COPYR STOPS |
| EOI ENCOUNTERED | COPYR | COPYR STOPS |
| EOI ENCOUNTERED-M FILES | COPYF | M-TH FILE ENDED WITH EOF, FOLLOWED BY EOI. COPYF STOPS. |
| EOI ENCOUNTERED-FILE M | COPYF | M-TH FILE ENDED WITH EOI, WITHOUT EOF. EOF ADDED TO FILOUT. COPYF STOPS. |

COPYRM(FILIN,FILOUT,N)

COPY AND CONVERT RECORDS ON SEQUENTIAL
(SQ) FILES FROM ONE RECORD TYPE AND BLOCK
STRUCTURE TO ANOTHER.

FILIN - INPUT FILE (NO DEFAULT)

FILOUT - OUTPUT FILE (NO DEFAULT)

N - A COUNT WHICH MAY BE EXPRESSED IN ANY OF THE FOLLOWING
FORMS (DEFAULT: 1P):

EOI - COPY TO END-OF-INFORMATION

MF - COPY M FILES (PARTITIONS)

MP - SAME AS MF

MR - COPY M SYSTEM LOGICAL RECORDS (SECTIONS)

MS - SAME AS MR

M - SAME AS MF

FILE CARDS ARE USED TO SPECIFY CONVERSION AND BLOCKING.
COPYRM IS ESPECIALLY CONVENIENT FOR READING AND WRITING BLOCKED
STRANGER TAPES (BOTH 7-TRACK AND 9-TRACK).
COPYRM IS VERY FAST FOR TAPE AND OTHER CONVERSION COPIES.

COPYSF(FILIN,FILOUT,N,PARAMS)

COPY (SHIFTED) FILES (MAY ALSO BE
CALLED COPYSBF)

COPYSR(FILIN,FILOUT,N,PARAMS)

COPY (SHIFTED) RECORDS

FILIN - INPUT FILE (MUST BE FIRST PARAMETER) (DEFAULT: INPUT)

FILOUT - OUTPUT FILE (MUST BE SECOND PARAMETER)

(DEFAULT: OUTPUT)

N - NUMBER OF FILES/RECORDS TO BE COPIED (DEFAULT: 1)
(USUALLY THIRD PARAMETER, BUT MAY APPEAR ELSEWHERE.)

PARAMS - OTHER PARAMETERS AND OPTIONS WHICH MAY FOLLOW IN ANY
ORDER. SOME OF THEM ARE:

SM - SHIFT M COLUMNS TO RIGHT (0-132) (DEFAULT: 1)

C - INITIAL RECORD IS CODED
(AUTOMATIC PARITY SWITCHING OCCURS, IF
NECESSARY, AT START OF EACH LOGICAL RECORD ON
TAPE FILES.)

EOI - COPY TO EOI FOR COPYSF/COPYSBF

EOF - COPY TO EOF FOR COPYSR

LC=L - LINE COUNT, L IS LINE LIMIT (DEFAULT: 2000)

LR - LIST END-OF-RECORD AND END-OF-FILE AS *EOR AND
*EOF, RESPECTIVELY. (DEFAULT: LR NOT SELECTED)

L8 - SELECT 8 LINES PER INCH OUTPUT
(RESTORED TO 6 LINES PER INCH AT END)

THIS IS AN ADAPTATION OF THE STANDARD CDC UTILITY *COPYSBF* WITH
MANY ADDITIONAL OPTIONS AND CAPABILITIES. IT IS USED PRIMARILY FOR
LISTING FILES BUT CAN BE USED AS A GENERAL PURPOSE UTILITY. (SEE
OCLIB/P: COPYSF)

SEE FULL DOCUMENTATION, IN OCLIB, FOR ADDITIONAL PARAMETERS.

*** SAMPLE UTILITY SETUPS ***

COPY BCD FROM FILE 3 TO NEW TAPE, FILE 1

| | |
|---|-------------------------------------|
| USERTAP,T200,MT2. | NAME / CODE |
| CHARGE,USER,JJJJJJJJJJ. | |
| LABEL,SGMAST,L=USERMASTER,W,D=HY,T=7,VSN=CA9999,RING. | |
| LABEL,SGSHIP,L=USERSHIPS,R,D=HY,VSN=CA8888,NORING. | |
| COPYF(SGSHIP,NULL,2,C) | COPY FIRST 2 FILES TO POSITION TAPE |
| RETURN,NULL. | |
| COPYF(SGSHIP,SGMAST,C) | COPY THIRD FILE TO A NEW TAPE |
| REWIND(SGSHIP,SGMAST) | WILL PRINT BLOCK COUNT WRITTEN |
| " 6/7/8/9 EOF | |

ANOTHER WAY TO SKIP AHEAD TWO FILES

| | |
|----------------------|---------------------------------|
| SKIPF(SGSHIP,2,17,C) | INSTEAD OF COPYF(SGSHIP,NULL,2) |
|----------------------|---------------------------------|

MULTIPLE INPUT TO FORTRAN PROGRAM

| | |
|---|---------------------------------|
| USERTP3,CM50000,T400,P4. | NAME / CODE |
| CHARGE,USER,JJJJJJJJJJ. | |
| COPYR(INPUT,TAPE2) | COPY CARDS TO FILE CALLED TAPE2 |
| REWIND,TAPE2. | REPOSITION TAPE2 FOR READING |
| FTN. | |
| LGO. | |
| " 7/8/9 EOR | |
| (DATA TO BE READ FROM TAPE2 IN PROGRAM) | |
| " 7/8/9 EOR | |
| PROGRAM MANIP (INPUT,OUTPUT,TAPE2,TAPE5=INPUT,TAPE6=OUTPUT) | |
| (FORTRAN SOURCE CARDS) | |
| " 7/8/9 EOR | |
| (DATA TO BE READ FROM TAPE5) | |
| " 6/7/8/9 EOF | |

TAPE TO PRINT - SINGLE SPACE

| | |
|--|-------------------------|
| USERPR,MT1,T100. | NAME / CODE |
| CHARGE,USER,JJJJJJJJJJ. | |
| LABEL(TAPE9,L=USERSOURCE,D=HY,R,VSN=CA9999,NORING) | |
| COPYSF,TAPE9,OUTPUT,C. | COPY SHIFTED CODED FILE |
| " 6/7/8/9 EOF | |

REPLACE SUBROUTINES IN RELOCATABLE BINARY PROGRAM

```

USERDTE,CM50000,T200,MT2.
CHARGE,USER,JJJJJJJJJJ.
FTN(P=BINARY,OPT=1)
REQUEST,NEWLIB,*PF.
ATTACH,OLDLIB,PROGBIN,ID=XXXX,PW=RDONLY.
REWIND,BINARY.
COPYL(OLDLIB,BINARY,NEWLIB)
NEWLIB.
CATALOG,NEWLIB,PROGBIN,ID=XXXX,PW=RDONLY,XR=RDONLY.
PURGE,OLDLIB.
* 7/8/9 EOR
  (SUBROUTINES TO REPLACE OLD VERSION)
* 7/8/9 EOR
  (DATA)
" 6/7/8/9 EOF

```

NAME / CODE

SEE CCRM, 4-17 FOR CREATE

EXECUTE CORRECTED FILE

AFTER CATALOG OF NEW CYCLE, PURGE OLD ONE

REPLACE, INSERT OR DELETE ROUTINES ON BINARY OBJECT FILE

```

...CARDS TO SET UP THE THREE FILES
COPYN(0,NEWLIB,OLDLIB,BINARY)
* 7/8/9 EOR
DECK1,DECKL,OLDLIB
PUTIN,,BINARY
DECKM,DECKN,OLDLIB
ADDCN,*,BINARY
* 7/8/9 EOR

```

** COPY FIRST L DECKS FROM OLDBIN FILE

** INSERT NEW ROUTINE PUTIN

** COPY LAST 2 DECKS FROM OLDBIN FILE

** ADD ON REST OF NEW ROUTINES FROM BINARY

SELECT ROUTINES FROM BINARY (OBJECT) FILE

```

USERSEL,CM50000.
CHARGE,USER,JJJJJJJJJJ.
ATTACH,BINLIB,ID=XXXX,MR=1.
REQUEST,GOFILF,*PF.
COPYN(0,GOFILF,BINLIB)
COMMENT. TWO ALTERNATIVE WAYS OF SELECTING A ROUTINE ARE SHOWN.
CATALOG,GOFILF,ID=XXXX,XR=VOIG.
* 7/8/9 EOR
EIGEN
HESS2,,BINLIB
WEOF(GOFILF)
* 7/8/9 EOR
" 6/7/8/9 EOF

```

NAME / CODE

COPY A RANDOM FILE

| | |
|----------------------|-------------------|
| JOBNAME,MT1,..... | NAME/CODE |
| CHARGE,..... | |
| ATTACH,RANIN,..... | INPUT RANDOM FILE |
| REQUEST,RANOUT,*PF. | OUTPUT FILE |
| COPYE,RANIN,RANOUT. | |
| CATALOG,RANOUT,..... | |
| " 6/7/8/9 EOF | |

CONVERT BLOCKED STRANGER TAPE (7-TRACK) TO CARD IMAGE DISK FILE

| | |
|---|-------------|
| JOBNAME,MT1,..... | NAME/CODE |
| CHARGE,..... | |
| VSN,TAPE=SLOT06=MYTAP1. | |
| REQUEST,TAPE,HY,S,NORING. | |
| REQUEST,DISK,*PF. | |
| FILE,TAPE,BT=E,RT=F,MBL=5120,CM=YES,MRL=80. | INPUT FILE |
| FILE,DISK,BT=C,RT=Z,MRL=80. | OUTPUT FILE |
| COPYRM,TAPE,DISK. | |
| CATALOG,DISK,..... | |
| " 6/7/8/9 EOF | |

CONVERT 9-TRACK EBCDIC TAPE TO CARD IMAGE DISK FILE

| | | |
|---|-----------|--------|
| JOBNAME,NT1,..... | NAME/CODE | |
| CHARGE,..... | | |
| VSN,TAPE=SLOT03=MYTAP2. | | |
| REQUEST,TAPE,HD,S,EB,NORING. | | |
| REQUEST,DISK,*PF. | | |
| COMMENT. SPECIFY DAYFILE MESSAGES AND | | |
| COMMENT. OPEN WITH REWIND, CLOSE WITH UNLOAD | | |
| FILE,TAPE,BT=E,RT=F,MRL=80,SDS=YES,CM=YES,OF=R,CF=U,MBL=5120. | | INPUT |
| COMMENT. REWIND BEFORE AND AFTER USE | | |
| FILE,DISK,BT=C,RT=Z,MRL=80,OF=R,CF=R. | | OUTPUT |
| COPYRM,TAPE,DISK. | | |
| CATALOG,DISK,..... | | |
| " 6/7/8/9 EOF | | |

CREATE 9-TRACK EBCDIC TAPE OF CARD IMAGES BLOCKED 20

| | |
|---|-----------|
| JOBNAME,NT1,..... | NAME/CODE |
| CHARGE,..... | |
| VSN,TAPE=SLOT04=MYTAP3. | |
| ATTACH,DISK,CARDIMAGES,ID=XXXX. | |
| REQUEST,TAPE,HD,S,EB,RING. | |
| FILE,DISK,BT=C,RT=7,MRL=80. | INPUT |
| FILE,TAPE,BT=K,RT=F,MRL=80,RB=20,MBL=1600,CM=YES. | OUTPUT |
| COPYRM,DISK,TAPE. | |
| RETURN,TAPE. | |
| " 6/7/8/9 EOF | |

***** LIBRARY MAINTENANCE UTILITIES *****

UPDATE IS A UTILITY FOR MAINTAINING LIBRARIES OF SOURCE PROGRAMS AND DATA. IT IS DESCRIBED ON PAGES 7-2 THRU 7-8.

EDITLIB IS A UTILITY FOR MAINTAINING LIBRARIES OF OBJECT PROGRAMS, BOTH RELOCATABLE AND ABSOLUTE. IT IS DESCRIBED ON PAGES 7-9 THRU 7-17.

BEGIN/REVERT IS A METHOD OF EXECUTING A SERIES OF FREQUENTLY USED CONTROL CARDS. THESE MAY BE GATHERED INTO LIBRARIES OF CATALOGUED PROCEDURES. BEGIN/REVERT AND CATALOGUED PROCEDURES ARE DESCRIBED ON PAGES 7-18 THRU THE END OF THE CHAPTER.

*** UPDATE ***

UPDATE IS A MAINTENANCE PROGRAM THAT CREATES, CORRECTS, AND MANIPULATES PROGRAM LIBRARY FILES.

UPDATE DATA MAY BE ANY SYMBOLIC INFORMATION (E.G., SOURCE CARDS FOR A COMPILER OR DATA CARDS). UPDATE IDENTIFIERS USUALLY DESTROY INFORMATION IN CARD COLUMNS 73-80. THE UPDATE LIBRARY IS A BINARY FILE OF CARD IMAGES IN COMPRESSED FORMAT. THE HISTORY INFORMATION RETAINED ON UPDATE FILES ALLOWS THE USER TO DELETE CORRECTIONS PREVIOUSLY MADE OR TO RESTORE FORMER (INACTIVE) CODE.

A DISK PROGRAM LIBRARY IS NORMALLY IN RANDOM FORMAT. THE RANDOM FILE (FASTER ACCESS BUT LARGER PRU SPACE) STORES EACH DECK AS A LOGICAL RECORD WITH DECK LIST, DIRECTORY, AND INDEX FOLLOWING THE LAST DECK. THE SEQUENTIAL FORMAT (REQUIRED FOR TAPE FILES) CONTAINS ONE BINARY RECORD IN WHICH DECK LIST AND DIRECTORY OCCUR FIRST.

UPDATE COMPILE FILES ARE 90-CHARACTER UNIT RECORDS UNLESS '8' OPTION WAS USED. SOURCE FILES AND 'UPDATE,8,....' COMPILE FILES ARE 80-CHARACTER UNIT RECORDS SUITABLE FOR PUNCHING.

*** UPDATE CARD PARAMETERS ***

FREQUENTLY USED FILE IDENTIFIERS

| LETTER | DEFAULT | PURPOSE |
|--------|---------|--|
| C | COMPILE | DEFAULT EXCEPT FOR B OPTION RUNS |
| C=0 | | DELETE CREATION OF COMPILE FILE |
| P | OLDPL | OLD PROGRAM LIBRARY FILE |
| N | NEWPL | MUST APPEAR IF NEW PL IS DESIRED |
| L | A1234 | SEVERAL SPECIAL PRINT OPTIONS EXIST |
| L=0 | | SUPPRESS UPDATE PRINTING |
| R | | SUPPRESS ALL REWINDS |
| | | OTHERWISE C,N,P,S REWIND BEFORE AND AFTER |
| I | INPUT | USE ONLY IF UPDATE DIRECTIVES ARE IN FILE |
| O | OUTPUT | USE ONLY IF UPDATE OUTPUT IS TO FILE |
| M | MERGE | USE ONLY TO COMBINE 2 OLD PL'S TO NEWPL |
| K | COMPILE | CAUSE COMPILE FILE TO CONTAIN DECKS IN COMPILE DIRECTIVE SEQUENCE - FOR OVERLAYS |
| S | SOURCE | FILE FOR UPDATE INPUT TO RESEQ DECKS |

FILE IDENTIFIERS FOR OLD PL AND NEW PL MAY NOT BE SAME FILE

OPTIONS ON WHAT GOES TO COMPILE FILE (UNLESS C=0)

F ALL DECKS (FROM OLD PL AND UPDATES)
 Q ONLY DECKS NAMED ON *COMPILE CARDS.
 WHEN Q AND N USED, SUBSET THESE TO NEWPL.
 DEFAULT WHEN F AND Q OMITTED: DECKS MODIFIED THIS RUN OR
 ON *COMPILE CARDS
 8 FORCE COMPILE FILE ENDPUNCHING INTO 73-80 FOR PUNCH

OPTIONS B, W DETERMINE IF NEWPL IS SEQUENTIAL OR RANDOM

B CONVERT RANDOM DISK OLDPL TO SEQUENTIAL NEWPL.
 (NO DIRECTIVES ARE READ AND NO COMPILE FILE CREATED)
 (USE TO BACKUP OLDPL TO TAPE)
 W DISK NEWPL IS TO BE SEQUENTIAL.

(SEE UPDATE, 4-2).

EXAMPLES: UPDATE(P,N)
 UPDATE,Q,P=LAST.

*** UPDATE DIRECTIVES ***

UPDATE REQUIRES AN INPUT LOGICAL RECORD OF DIRECTIVES (* IN COLUMN 1). A 7/8/9 EOR IS REQUIRED EVEN IF NO DIRECTIVES USED, EXCEPT WHEN PARAMETER A OR B IS USED ON THE CONTROL CARD.

| | |
|------------------------------------|--|
| *COMDECK NAME2 | STORE SET OF COMMON CARDS ONCE |
| *DECK NAME1 | *COMDECK MUST PRECEDE *DECKS THAT *CALL IT |
| *CALL NAME2 | INSERT NEW DECK (SUBPROGRAM OR GROUP) |
| | INSERT COMDECK NAME2 INTO CURRENT DECK |
| | HERE FOR COMPILE FILE |
| *ADDFILE FILEN,NAMEN | ADD FILES (INCLUDING *DECK CARDS) TO |
| | EXISTING LIBRARY AFTER DECK NAMEN |
| *ADDFILE | ADD INPUT DECKS WHICH FOLLOW, AFTER LAST |
| | DECK OF OLDPL |
| *IDENT UNIQUE | INITIATE CORRECTION SET |
| *INSERT NAMEI,SEQNO | INSERT CARDS AFTER SEQNO IN NAMEI |
| *DELETE NAMEI,SEQNO,NAMEI,SEQ2 | DELETE CARDS SEQNO TO SEQ2 INCLUSIVE |
| *DELFTE NAMEI,SEQNO,SEQ2 | AND OPTIONALLY INSERT OTHERS |
| *DELFTE NAMEI,SEQNO | DELETE CARD SEQNO AND OPTIONALLY INSERT |
| *COMPILE NAMEI | COPY SPECIFIC DECK TO COMPILE FILE |
| *READ FILIN | INSERT RECORD FROM FILIN WITHOUT REWINDING |
| *SEQUENCE DNAMEI | RESEQUENCE ACTIVE CARDS IN DECK DNAMEI |
| *CCPY DNAME,NAMEI,SEQNO,NAMEI,SEQ2 | COPY IN CARDS FROM DECK DNAME |
| | AT CURRENT INSERT OR DELETE POSITION |
| *RESTORE NAMEI,SEQNO | REACTIVATE A DELETED CARD AND OPTIONALLY |
| | INSERT OTHERS |
| *WEOR | CAUSE COMPILE FILE TO INCLUDE RECORD MARK |
| *YANK NAMEI | INACTIVATE A CORRECTION SET WITH ID NAMEI |
| *PURGE NAMEI | PERMANENTLY DELETE A CORRECTION SET NAME |
| *PURDECK NAMEI | PERMANENTLY DELETE ALL CARDS IN DECK NAMEI |
| *PULLMOD NAMEI | RECREATES A CORRECTION SET ON FILE SOURCE |
| *REWIND FILIN | REPOSITION FILIN AT START |
| *SKIP FILIN,N | SKIP FORWARD N LOGICAL RECORDS ON FILIN |
| | (A FILE MARK IS A LOGICAL RECORD ALSO) |

READ, SKIP, AND REWIND ARE NOT VALID FOR C,P,N,I,O,M,S FILES.

AN ALPHANUMERIC IDENTIFIER (MAXIMUM OF 9 CHARACTERS) GOES ON THE *DECK OR *COMDECK CARD FOR PROGRAM LIBRARY CREATION AND ON THE *IDENT CARD FOR CORRECTION OR EXPANSION RUNS. ALL CARDS IN A PARTICULAR DECK OR COMDECK ARE SEQUENTIALLY NUMBERED BY UPDATE (E.G., SC12.42 IS THE 42ND CARD WITH IDENTIFIER SC12. NOTE THAT CARD SC12.1 IS *DECK SC12). MOST DIRECTIVES WHICH MANIPULATE WHOLE DECKS OR IDENTs MAY CONTAIN A LIST (COMMA SEPARATED) OR AN INCLUSIVE LIST (PERIOD SEPARATED). THESE INCLUDE *COMPILE, *PURGE, *PURDECK, *SEQUENCE.

AFTER *ADDFILE DIRECTIVE ONLY COMPLETE DECKS MAY OCCUR. ANY FURTHER CORRECTION OR INSERTION CARDS MUST BE PRECEDED BY AN *IDENT.

MOST DIRECTIVES MAY BE ABBREVIATED (AT A TIME COST) BY 1 OR 2 LETTERS. PRINTED OUTPUT SHOWS FULL DIRECTIVE WITH ///// PRECEDING IT. FOR SUCCESSIVE CORRECTIONS IN SAME DECK THE *NAMEI.* MAY BE OMITTED.

SEVERAL COMPILE FILE DIRECTIVES ALLOW CONDITIONAL PROCESSING USING *IF, *END, *DEFINE.

*** SAMPLE UPDATE SETUPS ***

CREATE SOURCE PROGRAM TAPE LIBRARY, LIST

```

USERMAK,T200,MT1.                                NAME / CODE
CHARGE,USER,JJJJJJJJJJ.
LABEL,NEWPL,L=USERALONELIB,W,D=HY,T=7,VSN=CA9999,RING.
UPDATE(N,P)                ** CREATE NEW UPDATE PL, ALL DECKS TO COMPILE
RETURN,NEWPL.              ** RETURN TAPE DRIVE TO SYSTEM
COPYSF(COMPILF)            ** SINGLE SPACE LISTING WITH SEQUENCE NUMBERS
* 7/8/9 EOR
*DECK ALONE                UPDATE DIRECTIVES AND SOURCE DECKS
    (FORTRAN MAIN PROGRAM)
*DECK ATWO
    (SUBROUTINE)
*DECK THREE
    (SUBROUTINE)
*DECK FOUR
    (FUNCTION )
* 7/8/9 EOR
* 6/7/8/9 EOF

```

UPDATE OLD SOURCE LIBRARY TO NEW AND EXECUTE

```

USERDAE,T200,MT2.                                NAME / CODE
CHARGE,USER,JJJJJJJJJJ.
LABEL,OLDPL,L=USERALONELIB,R,D=HY,VSN=CA9999,NORING.
LABEL,NEWPL,L=USERTESLIB,D=HY,W,T=7,VSN=CA8888,RING.
UPDATE(N,F)                COPY PL WITH CORRECTIONS AND ALL TO COMPILE
RETURN,NEWPL,OLDPL.        RETURN TAPES TO SYSTEM
FTN(I)                     COMPILE FROM UPDATED FILE IMPLIES I=COMPILE
LGO.                       EXECUTE
* 7/8/9 EOR
*IDENT SG1030              CORRECTION MUST BE UNIQUE (INITIALS,DATE)
*INSERT ALONE,57           CORRECT DECK ALONE BY INSERT AFTER CARD 57
    (FORTRAN STATEMENTS)
*DELETE FOUR,12,13        CORRECT DECK FOUR REPLACING CARDS 12-13
    (NEW CARDS TO REPLACE DELETIONS - OPTIONAL)
* 7/8/9 EOR
* 6/7/8/9 EOF

```

ALTERNATIVE CARDS IF OLDPL AND NEWPL NAMES ARE CHOSEN BY USER

```

LABEL,SAM,L=USERTESLIB,D=HY,W,T=7,VSN=CA8888,RING.
REQUEST,JOE,HY,VSN=CA9999,NORING.
UPDATE(P=JOE,N=SAM,F)

```

CREATE SOURCE PROGRAM DISK LIBRARY, EXECUTE

```

USERB1,CM60000,T30.                                NAME / CODE
CHARGE,USER,JJJJJJJJJJ.
REQUEST,NEWPL,*PF.
UPDATE(N)
CATALOG(NEWPL,MASTPROGPL,ID=XXXX,XR=RDONLY)
CORCL(I=COMPILE)
LGC.
* 7/8/9      EOR
*DECK NAME1
  (CORCL SOURCE PROGRAM)
* 7/8/9      EOR
  (DATA CARDS, IF ANY)
" 6/7/8/9    EOF

```

CREATE BACKUP TAPE COPY OF DISK SOURCE LIBRARY

```

USERB2,T100,MT1.                                NAME / CODE
CHARGE,USER,JJJJJJJJJJ.
LABEL,NEWPL,L=USERMASTUP,T=30,D=HY,W,VSN=CJ6666,RING.
ATTACH,OLDPL,MASTPROGPL,ID=XXXX.
UPDATE(N,B)                                         CHANGE RANDOM FILE TO SEQUENTIAL
" 6/7/8/9    EOF

```

TOTAL REPLACEMENT OF EXISTING DECK

```

USERREP,MT1.                                NAME / CODE
CHARGE,USER,JJJJJJJJJJ.
LABEL,OLDPL,L=USERALONELIR,R,D=HY,VSN=CA9999,NORING.
REQUEST,NEWPL,*PF.
UPDATE(N,E)
CATALOG,NEWPL,ALONEPL,ID=USER,PW=KEEP,XR=KEEP.
* 7/8/9      EOR
*PURDECK THREE
*PURGE THREE
*ADDFILE
*DECK THREE
  (SUBROUTINE SOURCE CARDS)
" 6/7/8/9    EOF

```

SELECT ROUTINES OFF SOURCE SUBROUTINE LIBRARY
AND COMPILE WITH OWN PROGRAM

```

USERUP,CM100000,T300,MT1.                NAME / CODE
CHARGE,USER,JJJJJJJJJJ.
FTN.                                COMPILE OWN PROGRAMS
LABEL,LIBR,L=CLIBRARYUPD3,D=HY,R,VSN=CA0279,NORING.
UPDATE(P=LIBR,Q,L=0)
UNLOAD,LIBR.
FTN(I=COMPILE,L=0,OPT=1)    OMIT L=0 IF LISTING OF FORTRAN IS DESIRED
LGO.                        EXECUTE
'  7/8/9      EOR
    (OWN FORTRAN DECKS)
'  7/8/9      EOR
*COMPILE ARPLN1                SELECT DECKS FROM LIBRARY
*COMPILE AMKUTM
'  7/8/9      EOR
    (DATA CARDS, IF ANY)
"  6/7/8/9    EOF

```

ADD ROUTINES TO EXISTING LIBRARY TAPE - PRINT

```

USERMO,T100,MT2.                NAME / CODE
CHARGE,USER,JJJJJJJJJJ.
LABEL,OLDPL,L=XXXXLASTLIB,R,D=H,VSN=CA8888,NORING.
LABEL,NEWPL,L=XXXXNEXTLIB,W,D=HY,T=7,VSN=CA9999,RING.
UPDATE(N)
RETURN,OLDPL,NEWPL.
COPYSF(COMPILE,OUTPUT)          LIST ADDITIONS TO LIBRARY
'  7/8/9      EOR
*IDENT SG12017
*ADDFILE                        ** INSERT NEW DECKS AT END
*DECK NEWONE
    (SUBROUTINE SOURCE CARDS)
*DECK ANOTHR
    (SUBROUTINE SOURCE CARDS)
"  6/7/8/9    EOF

```

PUNCH SELECTED DECK

```

JOBNAME,....                NAME / CODE
CHARGE,....
ATTACH,OLDPL,....
UPDATE,8,Q.
ROUTE,COMPILE,DC=PU,TID=C.    PUNCH AT CENTRAL SITE
'  7/8/9      EOR
*COMPILE AMKUTM
"  6/7/8/9    EOF

```


JUNE 1977

PAGE 7-7

CREATE PROGRAM LIBRARY FROM STRANGER SOURCE FILE

| | NAME / CODE |
|--|--|
| USERSE,MT1. | |
| CHARGE,USER,JJJJJJJJJJ. | |
| VSN,TAPE1=SLOTXX=STRANG. | |
| REQUEST,NEWPL,*PF. | |
| REQUEST,TAPE1,S,HI,NORING. | |
| BEGIN,COPYBLK. | REBLOCK TO SI FILE - SEE CCLIB/P |
| RETURN,TAPE1. | RELEASE TAPE DRIVE |
| REWIND,TAPE2. | |
| UPDATE(P,N) | CREATE ONE DECK CONTAINING WHOLE PROGRAM |
| CATALOG,NEWPL,STRANGEPL,ID=USER,XR=KEEP. | |
| COPYSE,COMPILE. | LIST LIBRARY WITH ASSIGNED SEQUENCING |
| * 7/8/9 EOR | |
| *DECK PROG | |
| *READ TAPE2 | |
| " 6/7/8/9 EOF | |

SUBDIVIDE INTO DECKS - CORRECT

| | NAME / CODE |
|---|-------------------------------------|
| USERSD. | |
| CHARGE,USER,JJJJJJJJJJ. | |
| ATTACH,CLOPL,STRANGEPL,ID=USER. | |
| REQUEST,NEWLIB,*PF. | |
| UPDATE(P,N=NEWLIB) | |
| CATALOG,NEWLIB,STRANGEPL,ID=USER,XR=KEEP,PW=KEEP. | |
| FTN,I. | |
| LGO. | |
| * 7/8/9 EOR | |
| *IDENT SG07573 | |
| *INSERT PROG.1 | |
| PROGRAM ANY (INPUT,OUTPUT,TAPE8) | |
| CORRECTED FOR CDC FTN JAN 1973 S GOOD | |
| *INSERT PROG.2 | |
| *COPY PROG,PROG.5,PROG.6 | MOVE SPECIFICATION STATEMENTS |
| *DELETE PROG.5,6 | |
| *INSERT PROG.16 | |
| *DECK SUB1 | |
| *INSERT PROG.25 | |
| *DECK SUB2 | |
| *INSERT PROG.31 | |
| *DECK SUB3 | |
| *INSERT PROG.44 | |
| *DECK ZERO | |
| *SEQUENCE SUB1.ZERO | SEQUENCE EACH DECK (SUB1 THRU ZERO) |
| * 7/8/9 EOR | |
| (DATA CARDS) | |
| * 7/8/9 EOR | |
| " 6/7/8/9 EOF | |

SAMPLE STRANGER SOURCE FILE LIST

```
C  PROGRAM FROM OTHER COMPUTER SYSTEM
    DATA B/18*0./
    DATA BL,PI/1H , 3.14159/
    DIMENSION B(18),C(6)
    INTEGER BL,C
3  READ 2, IT,J,A
2  FORMAT(A5,I5,F10.2)
    IF(IT .EQ. BL) GO TO 9
    CALL SUB1 (IT,J,C,K)
    IF( N .NE. 0) CALL SUB2(IT,J,A)
    CALL SUB3(A,B,K)
    IF(K .EQ. 0) GO TO 3
    WRITE (8) C
9  STOP
    END
    SUBROUTINE SUB1(ALP,J,C,K)
    DIMENSION C(6)
    IF( J .GT. 6) GO TO 2
    C(J) = ALP
    K = 0
    RETURN
2  K = 1
    RETURN
    END
    SUBROUTINE SUB2(ALP,J,A)
C  ERROR PRINTOUT
    PRINT 2, ALP,J,A
2  FORMAT(12H0DATA ERROR  A6,I6,F12.2)
    RETURN
    END
    SUBROUTINE SUB3(A,B,K)
    DIMENSION B(18)
    DO 5 I=1,18
    IF(B(I) .EQ. 0.) GO TO 3
5  CONTINUE
    WRITE (8) B
    CALL ZERO (B)
    K = 1
    RETURN
3  B(I) = A
    K = 0
    RETURN
    END
    SUBROUTINE ZERO(B)
    DIMENSION B(18)
    DO 1 I=1,18
    B(I) = 0.
1  CONTINUE
    RETURN
C  USE THIS DECK AS INPUT TO PRECEDING EXAMPLES
    END
```

*** EDITLIB ***

THROUGH THE EDITLIB UTILITY PROGRAM, A USER CAN CREATE A LIBRARY OF BINARY ROUTINES. EDITLIB CAN HANDLE THE BINARY OUTPUT OF ANY COMPILER OR CORE IMAGE (ABSOLUTE) MODULES (SEE CCRM, 2-13). GENERALLY, THERE CANNOT BE MORE THAN ONE RELOCATABLE MAIN MODULE IN A LIBRARY; THERE CAN BE MORE THAN ONE CORE IMAGE MODULE. ABSOLUTE FILES CREATED BY SEGLOAD CANNOT BE PROCESSED BY EDITLIB.

RELOCATABLE MODULES ON A USER EDITLIB ARE SUITABLE FOR AUTOMATIC LOADING BY LOADER. A LIBRARY IS MADE AVAILABLE TO THE LOADER BY ONE OF THE FOLLOWING LOADER CONTROL CARDS:

LIBRARY,LIBNAME. WHERE 'LIBNAME' IS THE LFN OF THE LIBRARY
LDSET(LIB=LIBNAME)

EDITLIB CAN BE USED TO ADD, DELETE OR REPLACE ROUTINES, MODIFY SELECTED PARAMETERS, AND PROVIDE STATISTICS ABOUT THE LIBRARY CONTENTS. (SEE NCSBE, 4-35; SCOPE, CHAPTER 7; SUG, CHAPTER 7)

*** EDITLIB CONTROL CARD ***

THE FORMAT OF THE EDITLIB CONTROL CARD IS:

EDITLIB(I=LFNDIR,L=LFNLIST)

ALL PARAMETERS ARE OPTIONAL.

LFNDIR - FILE CONTAINING DIRECTIVES. (DEFAULT: 'INPUT')

LFNLIST - FILE TO CONTAIN LISTABLE OUTPUT. (DEFAULT: 'OUTPUT')

*** EDITLIB DIRECTIVES ***

EDITLIB DIRECTIVES APPEAR IN COLUMNS 1-72, ONE DIRECTIVE PER CARD AND HAVE ONE OF THE FOLLOWING FORMATS:

KEYWORD.

KEYWORD(<PARAMETERS>)

DIRECTIVES FALL INTO SIX CATEGORIES: CREATING A LIBRARY; MODIFYING A LIBRARY; MANIPULATING FILES; CHANGING LIBRARY FORMAT; LISTING STATISTICS; MISCELLANEOUS.

IN THE DESCRIPTIONS BELOW, <PROG> IS A ROUTINE NAME OR RANGE OF ROUTINES AND MAY HAVE ONE OF THE FOLLOWING FORMS:

| | |
|-------------------|--|
| * | ALL ROUTINES ON SPECIFIED FILE |
| PROG1 | SPECIFIC ROUTINE 'PROG1' |
| PROG1/PROG2/PROG3 | SPECIFIC ROUTINES 'PROG1', 'PROG2', 'PROG3' |
| PROG1+PROG2 | ROUTINES 'PROG1' THRU 'PROG2', INCLUSIVE |
| PROG1-PROG2 | ALL ROUTINES EXCEPT 'PROG1' THRU 'PROG2', INCLUSIVE |
| *+PROG2 | ALL ROUTINES FROM CURRENT POSITION THRU 'PROG2', INCLUSIVE |
| PROG1+* | ALL ROUTINES FROM 'PROG1' TO END OF LIBRARY |

** CREATING A LIBRARY **

LIBRARY(LIBNAME,NEW) MUST PRECEDE ALL OTHER DIRECTIVES EXCEPT
FILE MANIPULATION AND COMMENTS.

ADD(<PROG>,FROM) ADD SPECIFIED ROUTINE(S) OF FILE 'FROM'

FINISH. REQUIRED TO TERMINATE LIBRARY CREATION

** MODIFYING A LIBRARY **

LIBRARY(LIBNAME,OLD) MUST PRECEDE ALL OTHER DIRECTIVES EXCEPT
FILE MANIPULATION AND COMMENTS.

ADD(<PROG>,FROM) ADD SPECIFIED ROUTINE(S) OF FILE 'FROM'.
IF A ROUTINE ALREADY EXISTS, AN ERROR MSG
IS ISSUED AND THE ROUTINE IS NOT ADDED.

REPLACE(<PROG>,FROM) EXISTING ROUTINE(S) ARE REPLACED. IF A
ROUTINE DOES NOT EXIST, A MESSAGE IS ISSUED
AND THE ROUTINE IS ADDED TO THE LIBRARY.

DELETE(<PROG>) SPECIFIED ROUTINE(S) ARE DELETED.

SETAL(<PROG>,LEVEL) THE ACCESS LEVEL OF SPECIFIED ROUTINE OR
RANGE OF ROUTINES IS CHANGED TO 'LEVEL'.
TO BE ACCESSIBLE BY CONTROL CARD OR FROM
INTERCOM, 'LEVEL' MUST BE ODD. DEFAULT: 0
GENERALLY, USE AL=0 FOR SUBPROGRAMS;
AL=3 FOR PROGRAMS.

FINISH. REQUIRED TO TERMINATE LIBRARY MODIFICATION

** FILE MANIPULATION **

FILE MANIPULATION CARDS MAY APPEAR ANYWHERE IN THE DIRECTIVE
RECORD. AFTER AN EDITLIB, A RANDOM LIBRARY IS REWOUND; A SEQUENTIAL
LIBRARY IS AT END-OF-FILE.

REWIND(LFN) REWIND FILE LFN
REWIND(LFN1/LFN2/...LFNN) REWIND ALL FILES NAMED

SKIPF(N,LFN) SKIP FORWARD N DECIMAL RECORDS ON FILE LFN
SKIPF(<PROG>,LFN) SKIP FORWARD TO BEGINNING OF NAMED PROGRAM
ON SEQUENTIAL FILE LFN
SKIPF(N,LFN,F) SKIP FORWARD N DECIMAL FILES ON
MULTI-FILE LFN

SKIPB(N,LFN) SKIP BACK N DECIMAL RECORDS ON FILE LFN
SKIPB(<PROG>,LFN) SKIP BACK TO BEGINNING OF NAMED ROUTINE
ON SEQUENTIAL FILE LFN
SKIPB(N,LFN,F) SKIP BACKWARD N DECIMAL FILES ON
MULTI-FILE LFN

**** CHANGING LIBRARY FORMAT ****

A DISK RESIDENT LIBRARY IS IN RANDOM FORMAT; A TAPE RESIDENT LIBRARY IS SEQUENTIAL. TO CHANGE FORMATS, USE ONE OF:

| | |
|---------------------|--|
| RANTOSEQ(RLFN,SLFN) | DISK RESIDENT RANDOM LIBRARY, RLFN, IS CHANGED TO TAPE RESIDENT SEQUENTIAL LIBRARY, SLFN. |
| SEQTORAN(SLFN,RLFN) | TAPE RESIDENT SEQUENTIAL LIBRARY, SLFN, IS COPIED TO DISK LIBRARY, RLFN, WHICH WILL BE RANDOM. SLFN REWOUND AFTER EACH COPY. |

THESE DIRECTIVES MAY NOT APPEAR BETWEEN 'LIBRARY' AND 'FINISH'.

**** LISTING STATISTICS ****

A LIST OF INFORMATION ABOUT ANY OR ALL ROUTINES ON A LIBRARY FILE OR ANY BINARY FILE OF ROUTINES IS OBTAINED BY THE LISTLIB OR CONTENT DIRECTIVES. THE LIST INCLUDES:

- PROGRAM NAME
- DATE, TIME AND COMPILATION MACHINE
- ENTRY POINTS
- EXTERNAL REFERENCES
- ACCESS LEVEL (AL)
- LENGTH OF OBJECT DECK IN CM WORDS
- TYPE OF PROGRAM (RELOCATABLE OR ABSOLUTE)

A LIBRARY FILE IS LISTED BY:

LISTLIB(<PROG>,LFN)

ANY FILE OF ASSEMBLED ROUTINES IS LISTED BY:

CONTENT(<PROG>,LFN)

IN BOTH CASES, 'PROG' CAN HAVE ANY OF THE FORMS DESCRIBED ON 7-9.

'LISTLIB' AND 'CONTENT' DIRECTIVES MAY NOT APPEAR BETWEEN 'LIBRARY' AND 'FINISH'.

**** MISCELLANEOUS DIRECTIVES ****

ENDRUN. STOPS EXECUTION OF DIRECTIVES. NORMALLY THE LAST DIRECTIVE. ANY DIRECTIVES WHICH FOLLOW 'ENDRUN.' WILL BE CHECKED FOR SYNTAX BUT WILL NOT BE PROCESSED, NOR WILL ANY ERRORS IN THEM CAUSE THE JOB TO ABORT.

*/ IN COL 1-2 INDICATES A COMMENT AND MAY APPEAR ANYWHERE IN THE DIRECTIVE RECORD

*** EDITLIB EXAMPLES ***

CREATE DISK LIBRARY OF RELOCATABLE ROUTINES

| JOBNAME,..... | NAME/CODE |
|--------------------------------------|---------------------------------------|
| CHARGE,..... | |
| FTN(OPT=1) | ** COMPILE SUBROUTINES TO 'LGO' |
| REQUEST,MYLIB,*PF. | |
| EDITLIB. | ** CREATE USER LIBRARY |
| CATALOG,MYLIB,ID=XXXX,XR=XX. | ** PFN=MYLIB |
| " 7/8/9 EOR | |
| FORTRAN SOURCE FOR SUBPROGRAM 'SUB1' | |
| FORTRAN SOURCE FOR SUBPROGRAM 'SUB2' | |
| FORTRAN SOURCE FOR SUBPROGRAM 'SUB3' | |
| " 7/8/9 EOR | |
| LIBRARY(MYLIB,NEW) | ** ESTABLISH NEW LIBRARY FILE 'MYLIB' |
| ADD(*,LGO) | ** ADD ALL ROUTINES FROM LGO |
| FINISH. | ** TERMINATE LIBRARY CREATION |
| ENDRUN. | ** END EDITLIB |
| " 6/7/8/9 EOF | |

MODIFY LIBRARY

| JOBNAME,..... | NAME/CODE |
|--|--|
| CHARGE,..... | |
| FTN(OPT=1) | ** COMPILE ROUTINES TO GO INTO LIBRARY (OR OTHER LANGUAGE COMPILER) |
| ATTACH,MYLIB,ID=XXXX,PW=XX. | ** MUST HAVE EXTEND PERMISSION |
| EDITLIB. | ** USER EDITLIB |
| EXTEND,MYLIB. | ** USER MUST EXTEND LIBRARY |
| " 7/8/9 EOR | |
| NEW FORTRAN SOURCE FOR SUBPROGRAM 'SUB2' | |
| FORTRAN SOURCE FOR SUBPROGRAM 'SUB4' | |
| " 7/8/9 EOR | |
| LIBRARY(MYLIB,OLD) | ** SET TO MODIFY OLD LIBRARY 'MYLIB' |
| REPLACE(SUB2,LGO) | ** DELETE OLD COPY OF 'SUB2' FROM MYLIB, ADD NEW COPY FROM LGO |
| ADD(SUB4,LGO) | ** ADD NEW ROUTINE 'SUB4' FROM LGO |
| SETAL(MYPROG,3) | ** SET ACCESS LEVEL SO THAT 'MYPROG' CAN BE CALLED BY CONTROL CARD OR FROM INTERCOM |
| FINISH. | ** TERMINATE LIBRARY MODIFICATION |
| ENDRUN. | ** END EDITLIB |
| " 6/7/8/9 EOF | |

NOTE: A SINGLE DIRECTIVE REPLACE(*,LGO) COULD HAVE BEEN USED INSTEAD OF REPLACE(SUB2,LGO) AND ADD(SUB4,LGO). SINCE 'SUB2' EXISTS, IT WOULD BE REPLACED. SINCE 'SUB4' DOES NOT EXIST, IT WOULD BE ADDED TO THE LIBRARY.

CONDENSE A LIBRARY

DELETED AND REPLACED FILES ARE NOT PHYSICALLY REMOVED FROM A USER EDITLIB LIBRARY. THE USER SHOULD PERIODICALLY CREATE A NEW LIBRARY FROM THE OLD ONE TO REMOVE THE GARBAGE. MAKE SURE THE NEW LIBRARY IS VALID (SEE CCLIB/P: COPYLIB).

```

JOBNAME,..... NAME/CODE
CHARGE,.....
REQUEST,MYLIB1,*PF.
ATTACH,MYLIB,ID=XXXX.
EDITLIB.
CATALOG,MYLIB1,MYLIB,ID=XXXX,PW=XX.
' 7/8/9 EOR
LIBRARY(MYLIB1,NEW) ** ESTABLISH NEW LIBRARY FILE 'MYFILE'
ADD(*,MYLIB,LIB) ** ADD ALL ACTIVE ROUTINES FROM LIBRARY
                     'MYLIB'
FINISH. ** TERMINATE LIBRARY CREATION
ENDRUN. ** END EDITLIB
" 6/7/8/9 END-OF-FILE

```

NOTE: IF THE LIBRARY CONTAINS CORE IMAGE MODULES, 'SETAL(<PROG>,3)' IS REQUIRED AFTER THE 'ADD' DIRECTIVE. <PROG> IS DESCRIBED ON PAGE 7-9.

PUT AN ABSOLUTE MODULE INTO A LIBRARY, THEN EXECUTE

```

JOBNAME,..... NAME/CODE
CHARGE,.....
FTN(OPT=1) ** COMPILE PROGRAM
LOAD(LGO) ** LOAD PROGRAM FROM 'LGO'
NOGO,LFN. ** COMPLETE LOADING, PUT ABSOLUTE ON 'LFN'
REQUEST,MYLIB2,*PF.
EDITLIB. ** USER EDITLIB
CATALOG,MYLIB2,ID=XXXX,XR=XX.
LIBRARY,MYLIB2. ** ESTABLISH LIBRARY FOR LOADER TO SEARCH
NEWPROG. ** EXECUTE 'NEWPROG' FROM MYLIB2
' 7/8/9 EOR
PROGRAM NEWPROG (...)
(Rest of FORTRAN SOURCE)
' 7/8/9 EOR
LIBRARY(MYLIB2,NEW) ** ESTABLISH NEW LIBRARY FILE 'MYLIB2'
ADD(NEWPROG,LFN,AL=3) ** ADD PROGRAM 'NEWPROG' FROM FILE 'LFN'
FINISH. ** TERMINATE LIBRARY CREATION
ENDRUN. ** END EDITLIB
' 7/8/9 EOR
DATA CARDS, IF ANY, FOR NEWPROG
" 6/7/8/9 EOF

```

USE SUBPROGRAMS FROM A LIBRARY

```
JOBNAME,..... NAME/CODE
CHARGE,.....
FTN.                ** PROGRAM REFERENCES ROUTINE(S) ON 'MYLIB'
ATTACH,MYLIB,ID=XXXX.
LDSET(LIB=MYLIB)    ** ESTABLISH USER LIBRARY FOR NEXT LOAD
LGO.
*   7/8/9          EOR
    FORTRAN SOURCE FOR PROGRAM WHICH CALLS AT LEAST ONE OF
    SUB1, SUB2, SUB3, SUB4
*   7/8/9          FOR
    DATA CARDS, IF ANY
"   6/7/8/9        EOF
```

USING A PROGRAM FROM A LIBRARY

```
JOBNAME,..... NAME/CODE
CHARGE,.....
ATTACH,MYLIB2,ID=XXXX. ** ATTACH USER LIBRARY
LIBRARY,MYLIB2.        ** MAKE LIBRARY AVAILABLE TO SYSTEM
NEWPROG.               ** EXECUTE PROGRAM IN LIBRARY
*   7/8/9            END-OF-RECORD
    DATA CARDS, IF ANY
"   6/7/8/9          END-OF-FILE
```

*** FTN OVERLAY IN A LIBRARY ***

IN AN OVERLAY PROGRAM, 'OVERLAY' STATEMENTS SPECIFY THE LOCAL FILE NAME ON WHICH THE ABSOLUTE OVERLAY MODULE RESIDES. WHEN AN OVERLAY IS CALLED, THIS FILE IS SEARCHED UNTIL THE REQUESTED OVERLAY IS FOUND.

WHEN AN OVERLAY PROGRAM IS ADDED TO AN EDITLIB USER LIBRARY, THREE THINGS MUST BE TAKEN INTO CONSIDERATION:

- 1) ONCE AN OVERLAY PROGRAM HAS BEEN PUT INTO AN EDITLIB USER LIBRARY, THE OVERLAY FILE NAME MUST BE THE LIBRARY FILE NAME. THE SUBROUTINE 'OVLNAME', IN LIBRARY 'NSRDC', ALLOWS AN OVERLAY PROGRAM TO RUN, REGARDLESS OF THE LOCAL FILE NAME GIVEN TO THE FILE CONTAINING THE PROGRAM, WHETHER IT IS A LIBRARY OR NOT. 'OVLNAME' OBTAINS THE NAME OF THE FILE CURRENTLY BEING EXECUTED. THIS FILE NAME IS THEN USED IN THE 'CALL OVERLAY' STATEMENTS. (SEE EXAMPLE 1 BELOW; CCLIB/N: OVLNAME)
- 2) IF AN OVERLAY PROGRAM IS REPLACED IN AN EDITLIB LIBRARY, THE LIBRARY SHOULD ALSO BE CONDENSED (TO PHYSICALLY REMOVE THE ORIGINAL COPY OF THE PROGRAM). MANY PROGRAMS WILL NOT RUN UNLESS THIS IS DONE. (SEE EXAMPLE 2 BELOW; CCLIB/P: COPYLIB)
- 3) EVEN THOUGH THE OVERLAY FACILITY ALLOWS OVERLAYS OF A SINGLE PROGRAM TO RESIDE ON MORE THAN ONE FILE (FOR FASTER LOCATION AND LOADING), PROGRAMS IN AN EDITLIB USER LIBRARY WILL RESIDE ON ONE FILE. THEREFORE, THE 'CALL OVERLAY' STATEMENTS MUST ALL REFER TO THE SAME OVERLAY FILE NAME.

EXAMPLES ARE ON THE NEXT 2 PAGES.

EXAMPLE 1: PUT AN OVERLAY INTO A LIBRARY

```

JOBNAME,..... NAME/CODE
CHARGE,.....
FTN.                ** RELOCATABLE MODULES ARE ON LGO
ATTACH,NSRDC.
LDSET,LIB=NSRDC.
LOAD(LGO)
NOGO.                ** ABSOLUTE OVERLAYS ARE ON 'MYFILE'
REQUEST,MYLIB,*PF.   OR ATTACH,MYLIB,ID=XXXX.
EDITLIB.
CATALOG,MYLIB,ID=XXXX. OR EXTEND,MYLIB.
'      7/8/9      EOR
OVERLAY (MYFILE, 0, 0)
PROGRAM MYPROG (INPUT=128, OUTPUT=128, ...)
C
C THE FOLLOWING COMMON IS USED TO TRANSMIT 'OVLFIL' TO
C ALL SUBPROGRAMS OR OVERLAYS WHICH NEED IT
C
COMMON /OVLF/ OVLFIL
CALL OVLFIL (OVLFIL)
...
CALL OVERLAY (OVLFIL, 1, 0)
CALL OVERLAY (OVLFIL, 2, 0)
...
ENDRUN.
OVERLAY (MYFILE, 1, 0)
PROGRAM AA
COMMON /OVLF/ OVLFIL
...
CALL OVERLAY (OVLFIL, 1, 1)
...
RETURN
END
OVERLAY (MYFILE, 1, 1)
PROGRAM BB
...
RETURN
END
OVERLAY (MYFILE, 2, 0)
ETC.
'      7/8/9      EOR
LIBRARY(MYLIB,OLD)
ADD(*,MYFILE)
*/
*/ MAKE 0,0 OVERLAY CONTROL CARD CALLABLE
*/ FROM BATCH AND INTERCOM
*/
SETAL(MYPROG,3)
FINISH.
END
"      6/7/8/9      EOF

```


EXAMPLE 2: REPLACE AN EXISTING ABSOLUTE OVERLAY PROGRAM AND CONDENSE

```

JOBNAME,.....                               NAME/CODE
CHARGE,.....
FTN.                                           ** RELOCATABLE MODULES ARE ON LGO
ATTACH,NSRDC.
LDSET,LIB=NSRDC.
LOAD(LGO)
NOGO.                                           ** ABSOLUTE OVERLAYS ARE ON 'MYFILE'
ATTACH,MYLIB,ID=XXXX.
REQUEST,NEWLIB,*PF.
EDITLIB.
CATALOG,NEWLIB,MYLIB,ID=XXXX.
PURGE,MYLIB.                                   ** TO GET RID OF OLD LIBRARY (OPTIONAL)
'      7/8/9      EOR
  (FTN SOURCE PROGRAM)
'      7/8/9      EOR
*/
*/ REPLACE PROGRAM IN OLD LIBRARY
*/
LIBRARY(MYLIB,OLD)
REPLACE(*,MYFILE)
FINISH.
*/
*/ CREATE NEW, CONDENSED LIBRARY
*/
LIBRARY(NEWLIB,NEW)
ADD(*,MYLIB,LIB)
SETAL(MYPROG,3)                               REPEAT FOR ANY OTHER MAIN PROGRAMS
FINISH.
ENDRUN.
"      6/7/8/9      EOF

```

EXAMPLE 3: EXECUTE A PROGRAM FROM A LIBRARY

```

JOBNAME,.....                               NAME/CODE
CHARGE,.....
ATTACH,MYLIB,ID=XXXX.                         ** ATTACH USER LIBRARY
LIBRARY,MYLIB.                                ** MAKE LIBRARY AVAILABLE TO SYSTEM
MYPROG.                                       ** EXECUTE PROGRAM IN LIBRARY
'      7/809      EOR
  (DATA, IF ANY)
"      6/7/8/9      EOF

```

*** CATALOGUED PROCEDURES ***
(BEGIN/REVERT)

A CATALOGUED PROCEDURE FACILITY HAS BEEN OBTAINED FROM THE UNIVERSITY OF WASHINGTON. THIS FACILITY ALLOWS THE CREATION AND USE OF 'SUBROUTINES' OF CONTROL CARDS.

A PROCEDURE CONSISTS OF A SERIES OF CONTROL CARDS RESIDING ON SOME FILE. EXECUTION OF THAT PROCEDURE IS STARTED BY A 'BEGIN' CONTROL CARD. RETURN FROM A PROCEDURE IS CAUSED BY A 'REVERT.' CONTROL CARD WITHIN THE PROCEDURE. A PROCEDURE MAY INCLUDE NESTED CALLS TO OTHER PROCEDURES.

THE 'BEGIN' CARD, WHICH MUST BE CONTAINED IN ONE CARD AND MAY NOT BE CONTINUED, IS USED TO TRANSFER CONTROL TO A PROCEDURE:

BEGIN(,LFN) WHERE LFN IS THE LOCAL FILE CONTAINING THE PROCEDURE TO BE CALLED. THE USER MUST ATTACH OR OTHERWISE CREATE THE PROCEDURE FILE BEFORE IT IS CALLED, EXCEPT WHEN USING THE SYSTEM PUBLIC ACCESS PROCEDURE FILE (SEE CCRM, 7-21) OR WHEN THE PROCEDURE FILE HAS LOCAL FILE NAME 'PROFIL'.

THE 'REVERT.' CARD IS USED TO RETURN FROM A PROCEDURE. IT MUST BE PART OF THE PROCEDURE. IF A 'REVERT.' CARD IS NOT FOUND IN THE PROCEDURE, THE JOB WILL TERMINATE AT THE END OF THE PROCEDURE. AS IN A REGULAR CONTROL CARD SEQUENCE, A FATAL ERROR DURING THE EXECUTION OF A CARD IN A PROCEDURE WILL CAUSE THE PROCEDURE CARDS TO BE SKIPPED UNTIL AN 'EXIT' CARD OR AN END-OF-RECORD OR END-OF-FILE IS ENCOUNTERED. TO GUARANTEE RETURN FROM A PROCEDURE, A 'REVERT.' CARD MUST ALSO OCCUR AFTER THE 'EXIT' CARD. FOR EXAMPLE,

```
... (CONTROL CARDS TO BE EXECUTED)
REVERT.
EXIT. OR EXIT(S) OR EXIT(U)
... (ANY CONTROL CARDS TO BE EXECUTED IF A FATAL ERROR
    OCCURS DURING THE PROCEDURE)
REVERT.
```

WHEN 'BEGIN' IS USED FROM INTERCOM, CONTROL RETURNS TO THE USER WHETHER OR NOT A 'REVERT.' CARD IS ENCOUNTERED.

THE ABOVE IS A DESCRIPTION OF THE SIMPLEST USE OF BEGIN/REVERT TO EXECUTE AN UNNAMED PROCEDURE. THE CATALOGED PROCEDURE FACILITY ALSO SUPPORTS NAMED PROCEDURES ON SEQUENTIAL OR RANDOM FILES, PARAMETER SUBSTITUTION (BOTH POSITIONAL AND KEYWORD), CONTROL CARD SKIPPING, CREATION OF DATA RECORDS, PARAMETER SUBSTITUTION IN DATA RECORDS, CONTROL CARD DUPLICATION, SELECTIVE CONTROL CARD EXECUTION.

FOR A COPY OF THE FULL 27-PAGE BEGIN/REVERT DOCUMENTATION, SEE COLIB OR USE THE FOLLOWING:

BEGIN,UTILITY,,PROGDOC,OTHER,,BEGIN,OUTPUT.
ROUTE,OUTPUT,....

*** DUPLICATE CONTROL CARDS ***
(DUP/ENDUP)

'DUP' IS A UTILITY (OBTAINED FROM THE UNIVERSITY OF WASHINGTON) WHICH DUPLICATES CONTROL CARDS. ONE OR MORE CONTROL CARDS ARE DUPLICATED 0 TO 99 TIMES IN THE CONTROL CARD STREAM.

THE 'DUP' CONTROL CARD HAS THE FORM:

DUP(N,M) WHERE

N IS THE DUPLICATION NUMBER (THE NUMBER OF TIMES TO DUPLICATE: 0 TO 99)

M IS THE NUMBER OF CARDS TO DUPLICATE (1 TO 99 OR *) (DEFAULT: 1)

IF N=0, M CARDS ARE SKIPPED.

IF M=*, DUPLICATION CONTINUES UNTIL AN 'ENDUP.' CONTROL CARD. (AN 'ENDUP.' CARD IS IGNORED IF ENCOUNTERED WHILE NOT DUPLICATING CONTROL CARDS.)

OCCURRENCES OF THE CHARACTER '\$' IN THE DUPLICATED CONTROL CARD ARE REPLACED WITH THE CURRENT DUPLICATION NUMBER (SEE EXAMPLE BELOW). THE LENGTH OF A CONTROL CARD MAY INCREASE WITH SUBSTITUTION OF THE DUPLICATION NUMBER. CONTROL CARDS GREATER THAN 80 CHARACTERS MAY BE ILLEGAL.

THE DUPLICATED CONTROL CARDS ARE LISTED ONCE IN THE DAYFILE WITH A LEADING * BEFORE BEING EXECUTED.

EXAMPLE

...
DUP(3,2)
COPYR(TAPE,TAPES)
REWIND(TAPES)

EXPANDS TO

...
COPYR(TAPE,TAPE1)
REWIND(TAPE1)
COPYR(TAPE,TAPE2)
REWIND(TAPE2)
COPYR(TAPE,TAPE3)
REWIND(TAPE3)
...

'DUP' SHOULD BE USED ONLY IN A CATALOGUED PROCEDURE. RESULTS ARE UNPREDICTABLE IF USED IN THE MAIN CONTROL CARD STREAM.

*** RANDOM PROCEDURE FILES ***

UNNAMED PROCEDURES MUST RESIDE IN A SEQUENTIAL FILE. NAMED PROCEDURES MAY RESIDE IN A SEQUENTIAL OR A RANDOM FILE. IF A FILE CONTAINS MANY NAMED PROCEDURES, SIGNIFICANT TIME CAN BE SAVED IN THE LOCATION OF A PROCEDURE IF A RANDOM PROCEDURE FILE IS USED. 'CPINDEX' READS A SEQUENTIAL FILE OF CATALOGUED PROCEDURES AND WRITES A RANDOM FILE WITH A NAME INDEX. IT IS EXECUTED AS FOLLOWS:

CPINDEX(SEQFILE,RANFILE)

'CPINDEX' HANDLES UP TO 200 PROCEDURES PER FILE AND PUTS THE NUMBER OF PROCEDURES INTO THE DAYFILE.

BE SURE TO SAVE A COPY OF THE SEQUENTIAL FILE AS THERE ARE MANY PROCEDURES FOR LISTING AND MANIPULATING SEQUENTIAL PROCEDURE FILES. (SEE CCLIB/P: PROADD/PROALL/PRODELE/PROGET/PROHDR/PROLIST/PRONAM/PROREPL)

*** PUBLIC-ACCESS PROCEDURES - PROFIL ***

'PROFIL' IS A LIBRARY OF PROCEDURES WHICH MAY BE EXECUTED BY USE OF THE 'BEGIN' COMMAND.

IT IS NOT NECESSARY TO ATTACH THIS FILE BEFORE USING ONE OF THE PROCEDURES. WHEN THE SECOND PARAMETER OF THE 'BEGIN' CARD IS OMITTED (E.G., BEGIN,PROCNAME,,<PARAMS>.), THIS FILE IS AUTOMATICALLY ATTACHED. THE USER MUST NOT USE 'PROFIL' AS THE LFN OF ANY OTHER FILE, OR IT WILL BE USED INSTEAD OF THE SYSTEM FILE.

WHEN THE SECOND PARAMETER OF THE 'BEGIN' CARD IS OMITTED AND ONLY KEYWORD PARAMETERS ARE USED (E.G., PROCEDURE 'AUDIT'), THEN THE SECOND COMMA OF THE '.,' MAY BE OMITTED (E.G., 'BEGIN,AUDIT,ID=XXXX.').

SEE CCLIB FOR A LIST OF THE AVAILABLE PROCEDURES AND CCLIB/P FOR FULL DOCUMENTATION OF EACH PROCEDURE.

SELECTED PROCEDURES ARE DESCRIBED IN DETAIL ON THE NEXT PAGE.

AUDIT

SORTED USER AUDIT

TO USE - BEGIN,AUDIT,KEYWORD-PARAMETERS.
KEYWORD-PARAMETERS ARE THOSE DESCRIBED ON
CCRM, PAGE 3-4 (AI=, AC=, ID=, LF=) PLUS
TYPE=STATIC - STATIC AUDIT (DEFAULT: DYNAMIC)

MAX FL - 350008

DEFAULT - BEGIN,AUDIT,AI=P,ID=XXXX,LF=OUTPUT. (BATCH)
BEGIN,AUDIT,AI=I,ID=XXXX,LF=OUTPUT. (INTERCOM)

EXAMPLE - BEGIN,AUDIT,AI=P,ID=XXXX.
(USE AUDIT BY ID BECAUSE IT IS FASTER AND
MUCH CHEAPER)

UTILITY

EXECUTE A PROGRAM ON EDITLIB USER LIBRARY 'UTILITY'

TO USE - BEGIN,UTILITY,,PROG,PARAMS.
PROG - PROGRAM TO BE EXECUTED
PARAMS - UP TO 19 PARAMETERS FOR 'PROG'

MAX FL - DEPENDS ON PROGRAM BEING EXECUTED

DEFAULT - THERE IS NO MEANINGFUL DEFAULT

JUNE 1977

PAGE 7-22.1

CCRM

PRINT COPIES OF COMPUTER CENTER REFERENCE MANUAL

TO USE - BEGIN,CCRM,,COPIES,OUTPUT,PM,CHAPTERS.

COPIES - NUMBER OF COPIES TO PRINT
(DEFAULT: 1)

OUTPUT - OUTPUT FILE

PM - IF NOT OMITTED, PRINT MESSAGES ARE
GENERATED FOR 1-PART, NARROW
UNLINED PAPER (FOR REMOTE
TERMINALS ONLY).
IF OMITTED, NO PRINTER MESSAGES
ARE GENERATED (CENTRAL SITE)
(DEFAULT)

CHAPTERS - ONE OF THE FOLLOWING:

ALL - PRINT ENTIRE MANUAL.
(DEFAULT)ONE OR MORE CHAPTER NUMBERS
SEPARATED BY COMMAS (0-15, WHERE
0 IS THE TABLE OF CONTENTS AND
15 IS THE INDEX)

MAX FL - 20000R

DEFAULT - BEGIN,CCRM,,1,OUTPUT,,ALL.

EXAMPLE - TO PRINT 1 COPY OF ENTIRE MANUAL ON NARROW,
UNLINED PAPER AT CENTRAL SITE:

BEGIN,CCRM.

ROUTE,OUTPUT,DC=PR,FC=1T,TID=C,FID=*.

TO PRINT 1 COPY OF CHAPTERS 2, 3, 6, 10:

BEGIN,CCRM,,,,,2,3,6,10.

TO PRINT 2 COPIES OF CHAPTER 4 (FORTRAN):

BEGIN,CCRM,,2,,,4.

TO PRINT 1 COPY OF MANUAL ON NARROW, UNLINED
PAPER AT REMOTE TERMINAL AE:

BEGIN,CCRM,,,OUT,PM.

ROUTE,OUT,DC=PR,TID=AE,FID=*.

JUNE 1977

PAGE 7-22.2

GRIPE

ALLOW USER TO MAKE GRIPEs OR SUGGESTIONS DIRECTLY TO THE COMPUTER.

to USE - BEGIN,GRIPE,,OUTPUT,PROMPT,WHOM,INPUT.
 OUTPUT - LISTABLE OUTPUT, IF USER REQUESTS A COPY OF HIS COMMENTS. TO HAVE THIS PRINT AT THE TTY, USER MUST CONNECT,OUTPUT.
 PROMPT - INTERACTIVE USE ONLY
 IF NO - ONLY MINIMAL PROMPTING IS PROVIDED.
 IF OMITTED OR ANYTHING ELSE - FULL PROMPTING IS PROVIDED.
 WHOM - RESERVED FOR FUTURE USE.
 INPUT - BATCH: FILE FROM WHICH GRIPE AND ADMINISTRATIVE INFORMATION IS TO BE READ.
 INTERACTIVE: IF GRIPE IS TO BE ENTERED DIRECTLY FROM THE KEYBOARD, OMIT THIS PARAMETER. IF THE GRIPE WAS PREPARED EARLIER (E.G., USING EDITOR), ENTER THE LFN OF THE FILE CONTAINING THE GRIPE (DO NOT USE LFN 'INPUT').

MAX FL - 360008

DEFAULT - INTERACTIVE: BEGIN,GRIPE,,OUTPUT.
 BATCH : BEGIN,GRIPE,,OUTPUT,,,INPUT.

DATA - BATCH USE ONLY
 THE GRIPE IS ENTERED IN COLUMNS 1-70 OF AS MANY CARDS AS REQUIRED (PLEASE BE BRIEF). THIS IS FOLLOWED BY A CARD WITH 'END' IN COLUMNS 1-3.
 THE LAST CARD HAS ADMINISTRATIVE INFORMATION FOR PREPARING A TROUBLE FORM, SHOULD ONE BE NECESSARY. IT CONTAINS:
 COL- CONIENIS
 1-20 NAME
 21-40 DTNSRDC CODE OR COMPANY NAME
 41-50 TELEPHONE
 51-54 USER INITIALS (OF THE FORM XXXX)

EXAMPLE - THE SIMPLEST FORM FOR INTERACTIVE USE:
 CONNECT,OUTPUT.
 BEGIN,GRIPE.

THE SIMPLEST FORM FOR BATCH USE:

| JOBNAME,.... | NAME/CODE |
|------------------------------|-----------|
| CHARGE,.... | |
| BEGIN,GRIPE. | |
| ' 7/8/9 EOR | |
| (GRIPE) | |
| END | |
| NAME ... CODE ... PHONE INIT | |
| " 6/7/8/9 EOF | |

***** INTERACTIVE GRAPHICS *****

ON THE CDC 6000 COMPUTERS AT DTNSRDC, INTERACTIVE GRAPHICS IS SUPPORTED BY THE INTERACTIVE GRAPHICS SYSTEM (PAGES 8-1 THRU 8-7) AND THE TEKTRONIX DISPLAY TERMINALS (PAGE 8-8).

*** INTRODUCTION TO IGS ***

THE DESIGN GOAL OF THE INTERACTIVE GRAPHICS SYSTEM (IGS) IS TO PERMIT REAL TIME USE OF A LARGE COMPUTER (CDC 6000) BY A GRAPHICS CONSOLE USER WITHOUT SIGNIFICANTLY DEGRADING THE CAPABILITIES OF THE LARGE COMPUTER.

IGS USES A SMALL COMPUTER (CDC 1700) TO CONTROL THE BASIC FUNCTIONS OF THE GRAPHICS HARDWARE. THE IGS SYSTEM USES THE 6700 TO HANDLE ONLY THE MORE DIFFICULT MANIPULATIONS AND TO DO THE MATHEMATICAL WORK REQUIRED BY THE APPLICATIONS PROGRAM.

THE DIGIGRAPHICS 274 DISPLAY CONSOLES CONNECTED TO THE CDC 1700 PERMIT THE USER TO CREATE, DISPLAY, STORE, RETRIEVE, AND MODIFY ANY GRAPHICS FORMS NECESSARY FOR THE ACTIVE ANALYSIS OF A PROBLEM, AS WELL AS GIVING HIM A MEANS OF ENTERING DATA DIRECTLY. THESE GRAPHIC FORMS CAN THEN BE EXPANDED OR CHANGED BY THE USER IN A REAL-TIME ENVIRONMENT THROUGH HIS APPLICATION PROGRAM AND THE INTERACTIVE GRAPHICS SYSTEM.

THE IGS PACKAGE ON THE DTNSRDC 6700 COMPUTER SERVES AS THE BASIS FOR ALL OTHER INTERACTIVE GRAPHIC SUBROUTINE PACKAGES CURRENTLY AVAILABLE.

REFERENCES:

| | |
|--|---------------------|
| 274 INTERACTIVE GRAPHICS SYSTEM, VERSION 2 | PUB. NO. 17303600 F |
| 274 INTERACTIVE GRAPHICS SYSTEM, NSRDC | PUB. NO. 17326700 A |

ADDITIONAL INFORMATION ABOUT DTNSRDC INTERACTIVE GRAPHIC SOFTWARE MAY BE OBTAINED BY CONTACTING:

USER SERVICES
MEL HAAS

CODE 1892.1
CODE 1843

JUNE 1977

PAGE 8-2

*** DTNSRDC GRAPHICS ***

CHANGES TO MANUAL

AEXEC MUST BE FOLLOWED BY AN AETSKC CALL TO THE NEXT OVERLAY TO BE EXECUTED.

ROUTINE GUMESG IS NOW SPELLED GIMESG.

CALLING SEQUENCE FOR DMINIT HAS ONLY THREE ARGUMENTS.
IF DMTRL IS USED, IT MUST BE IN 0,0 OVERLAY.

CONVERTING TASKING

REPLACE THE USE OF GIMAIN IN ZERO OVERLAY BY:
CALL AETSKC(NLAAAA) WHERE

N - NUMBER OF LETTERS IN AAAA, MAX 7
AAAA - NAME OF LEVEL ONE OVERLAY PROG.

GENERAL COMMENTS

SIGNON AT DTNSRDC IS AS DESCRIBED IN CHAPTER 8 OF 17326700 (SEE CCRM, 8-2: REFERENCES), NOT AS IN CHAPTER 6 OF 17303600.

IGS JOBS MUST BE OVERLAYED OR SEGMENTED. LOADER AND EDITLIB MAY BE USED TO CREATE THE TASK FILE AS BELOW. (SEE ALSO PRELOAD AND TSKLOAD CCRM, 8-4.)

*** IGS EDITLIB EXAMPLES ***

CREATE IGS TASK FILE FOR 274 DISPLAY

| | NAME / CODE |
|-----------------------------------|--------------------------------------|
| JORNAME,..... | |
| CHARGE,..... | |
| REQUEST,NEWPL,*PF. | |
| ATTACH,OLDPL,PFN,ID=XXXX. | |
| UPDATE,N,F,C=A. | |
| FTN,I=A,B=PROG. | COMPILE USER PROGRAM |
| CATALOG,NEWPL,PFN,ID=XXXX. | |
| LDSET(LIB=IGS274) | LOAD SYSTEM LIBRARY FOR IGS ROUTINES |
| LOAD,PROG. | LOAD USER BINARY FILE |
| NOGO. | |
| REQUEST,TASKFIL,*PF. | |
| EDITLIB. | |
| CATALOG,TASKFIL,PFN1,ID=XXXX. | |
| ' 7/8/9 EOR | |
| (UPDATE CORRECTION CARDS, IF ANY) | |
| ' 7/8/9 FOR | |
| LIBRARY(TASKFIL,NEW) | DEFINE TASKFIL FOR LIBRARY |
| REWIND(SOURCE) | SOURCE IS NAME OF ZERO LEVEL OVERLAY |
| ADD(*,SOURCE) | |
| FINISH. | |
| ENDRUN. | |
| " 6/7/8/9 EOF | |

ACCESS TASK FILE FOR EXECUTION

| | NAME / CODE |
|------------------------------|--------------------------------------|
| JORNAME,..... | |
| CHARGE,..... | |
| ATTACH,TASKFIL,PFN1,ID=XXXX. | |
| LIBLOAD(TASKFIL,CREATE) | CREATE IS NAME OF LEVEL ZERO PROGRAM |
| EXECUTE. | |
| ' 7/8/9 FOR | |
| TASKFIL | DO NO USE PERIOD AFTER THIS NAME |
| " 6/7/8/9 EOF | |

*** IGS UTILITIES ***

SEVERAL IGS UTILITIES HAVE BEEN DEVELOPED AT DTNSRDC BY THE GRAPHICS SYSTEM DEVELOPMENT GROUP. THOSE UTILITIES WHICH ALSO APPLY TO BATCH AND INTERCOM PROGRAMS ARE DENOTED WITH AN ASTERISK (*).

- * PRELOAD: AN DTNSRDC-DEVELOPED CONTROL CARD UTILITY FOR BUILDING, MAINTAINING, AND UTILIZING PROGRAM LIBRARIES IN RELOCATABLE BINARY FORMAT. EXTERNAL REFERENCES MAY BE SATISFIED FROM UP TO FIVE USER LIBRARIES. PRELOAD SUPPLIES ALL NEEDED ROUTINES FROM EACH LIBRARY INTO EACH OVERLAY. ALLOWS FOR REPLACEMENT OF INDIVIDUAL OBJECT ROUTINES WITHOUT EXTENSIVE RECOMPILATION. (SEE CMD-51-72) ATTACH,PRELOAD,CAMVPRELOAD,ID=CAMV.
- * TSKLOAD: AN DTNSRDC-DEVELOPED CONTROL CARD UTILITY FOR BUILDING AND MAINTAINING PROGRAM LIBRARIES IN ABSOLUTE FORMAT. TASKING IS A SET OF LIBRARY SUBROUTINES AND CONTROL CARD PROGRAMS TO PERMIT USE OF CALL-BY-NAME OVERLAYS (TASKS). THE DESIGN, CREATION, AND USE OF TASKS AIDS IN EFFICIENT UTILIZATION OF CORE MEMORY, MODULARIZES PROGRAMMING, AND SIMPLIFIES PROGRAM MAINTENANCE. FIELD LENGTH WILL BE AUTOMATICALLY REDUCED TO THE MINIMUM NEEDED FOR EACH OVERLAY. ATTACH,TSKLOAD,CAMVTSKLOAD,ID=CAMV.
- * INTERACTIVE DATA MANAGER: AN DTNSRDC-DEVELOPED SUBROUTINE PACKAGE WHICH PROVIDES AN EFFICIENT, GENERAL PURPOSE, VIRTUAL-MEMORY, DATA MANAGEMENT FACILITY. (SEE CMD-48-72) THE PRELOAD LIBRARY IS "CAMVNEWPRE".
- * EDIT AN DTNSRDC-DEVELOPED CONTROL CARD UTILITY TO SIMPLIFY AND AUTOMATE THE USE OF UPDATE, PRELOAD, AND TSKLOAD. A SINGLE EDIT CONTROL CARD MAY REPLACE UP TO 23 CARDS. (SEE CMD-24-73) ATTACH,EDIT,CAMVEDIT,ID=CAMV.
- AUTOEDIT: ADAPTED FROM THE AIR FORCE CAMBRIDGE RESEARCH LABORATORIES THIS INTERACTIVE GRAPHICS PROGRAM MAY BE USED TO EXAMINE, EDIT, CURVE FIT, AND PLOT DATA. (SEE CMD-50-72)

GIPLCT A SUBROUTINE PACKAGE AND ASSOCIATED POST-PROCESSING PROGRAMS TO PERMIT SELECTED ITEMS ON THE 274 DISPLAY SCREEN TO BE PLOTTED ON ANY OF THE HARDCOPY DEVICES (SC 4060, CALCOMP 936, CALCOMP 763, 1700 CALCOMP, OR COMPUTEK 400). FOR DETAILS CALL MEL HAAS, CODE 1843, (202) 227-1933.

274CALCOMP A PLOT SUBROUTINE TO ALLOW ALL THE CALCOMP SUPPORT LIBRARY TO PLOT ON THE 274 DISPLAY SCREEN. FOR DETAILS SEE MEL HAAS, CODE 1843, (202) 227-1933.

ITIS.2 THE INTERACTIVE TERMINAL INTERFACE SYSTEM, VERSION 2, PROVIDES THE FORTRAN APPLICATION PROGRAMMER WITH A FLEXIBLE INTERFACE TO THE 274. ALL ROUTINES ARE FORTRAN CALLABLE AND ARE USED PRIMARILY FOR TWO OR THREE DIMENSIONAL GRAPHICS DESIGN APPLICATIONS. TO ACCESS ITIS.2 SOFTWARE:
 ATTACH,ITIS,ITISEDITLIB,IO=SNJC.
 LDSET(LIB=IGS274/ITIS)
BEFORE LOAD OR LGO. (SEE CCRM, 8-3)

COPYFUL
COPY274 SUBROUTINES IN SYSTEM LIBRARY IGS274 FOR 274 INTERFACE. THE PACKAGE READS THE 1744 BUFFER MEMORY, INTERPRETS THE CODE, AND CALLS THE APPROPRIATE CALCOMP ROUTINES (SYMBOL, LINE, CIRCL) TO PROVIDE A COPY OF THE 274 CONSOLE SCREEN IMAGE. THE USER MUST ATTACH THE DESIRED CALCXXX PACKAGE (XXX IS ONE OF 170, 763, 936), AND MUST USE
 LDSET,LIB=IGS274/CALCXXX.
WHEN XXX IS 170, THE CALCOMP FILE PLOT FOR 1700 IS CREATED. WHEN XXX IS 936 OR 763, THE USER MUST
 CALL PLOTS (IBUF, NLOC, NT)
BEFORE USING THE PACKAGE (NT IS LESS THAN 100). THE OUTPUT FILE "TAPENT" MAY NOT BE A MAGNETIC TAPE BUT MAY BE CATALOGED FOR A LATER JOB TO COPY TO TAPE FOR PROPER CALCOMP PROCESSING. "TAPENT" MAY NOT BE USED FOR ANY OTHER PURPOSE IN THE USER'S PROGRAM.

*** SIGNON ***

'SIGNON' IS A UTILITY PROGRAM WHICH PROVIDES THE GRAPHICS USER WITH THE CAPABILITY OF BUILDING AND EXECUTING BATCH JOBS FROM THE IGS-274 CONSOLE.

IGS-274 CONSOLE ID'S ARE:

| | |
|---------|------------|
| DTNSROC | 1 |
| NAVSEC | 21 (OCTAL) |
| NAVAIR | 31 (OCTAL) |

SIGNON IS INITIATED/TERMINATED BY THE FOLLOWING:

| | |
|--------------------------|--------------------|
| CNTRL/G (KEYED TOGETHER) | - START SIGNON |
| CNTRL/E, THEN CNTRL/K | - TERMINATE SIGNON |
| CNTRL/K | - ABORT SIGNON |

SIGNON REQUESTS ENTRY OF A CHARGE CARD. AFTER THE JOB ORDER NUMBER IS VALIDATED, THE USER CAN BUILD A SET OF CONTROL CARDS. THIS CONTROL CARD BUFFER EXISTS FOR THE DURATION OF THE CURRENT CONSOLE SESSION. IT MAY NOT BE SAVED FOR USE AT A LATER TIME.

THE FOLLOWING OPTIONS ARE SELECTABLE VIA KEYBOARD OR LIGHT PEN. ALL KEYED ENTRIES ARE TERMINATED BY THE CARRIAGE RETURN (CR) KEY.

CLEAR CONTROL CARD BUFFER

RESTRUCTURE THE CURRENT CONTROL CARD BUFFER. ACTIVE CONTROL CARDS ARE RENUMBERED. THE USER MAY THEN MODIFY THE NEW BUFFER. TO COMPLETELY ERASE THE BUFFER, USE 'TERMINATE' OR DELETE EACH LINE IN THE CURRENT BUFFER AND THEN ENTER CLEAR.

PROMPT DISPLAY PROPER FORMATS FOR ENTERING COMMANDS.

LAST CONTROL CARD BUFFER

DISPLAY THE LAST BUFFER. THE BUFFER BEING BUILT WILL NOT BE DESTROYED.

DAYFILE BUFFER

DISPLAY THE LAST DAYFILE BUFFER.

EXECUTE EXECUTE THE CONTROL CARD BUFFER JUST BUILT OR MODIFIED.

TERMINATE END THE CONSOLE SESSION. MAY BE USED AT ANY TIME. DOES NOT TERMINATE SIGNON. THE SAME OR ANOTHER USER MAY CONTINUE BY ENTERING ANOTHER CHARGE CARD. THE CONTROL CARD BUFFER WILL BE EMPTY.

BUILD A CONTROL CARD SET

TO BUILD A SET OF CONTROL CARDS, SIMPLY TYPE EACH CONTROL CARD.
SIGNON WILL PREFIX EACH WITH A LINE NUMBER.

CORRECT (MODIFY) A CONTROL CARD SET

INSERT A CARD - TYPE LINE NUMBER, EQUAL SIGN, CONTROL CARD:
21=LDSET,LIB=NSRDC.

DELETE A CARD - TYPE LINE NUMBER, EQUAL SIGN:
20=

REPLACE A CARD - TYPE LINE NUMBER TO BE REPLACED, EQUAL SIGN, NEW
CONTROL CARD:
21=LDSET,LIB=NSRDC/CALC936.

ADD A CARD - TYPE CONTROL CARD WITHOUT LINE NUMBER. IT WILL BE
ADDED AT THE END OF THE CONTROL SET.

DELETE A CHARACTER - TYPE SHIFT/O (LETTER OH) OR SELECT 'BKSPACE' WITH
LIGHT PEN.

*** TEKTRONIX 4015 ***

THE TEKTRONIX 4015 IS A SYNCHRONOUS COMPUTER DISPLAY TERMINAL CAPABLE OF DISPLAYING BOTH ALPHANUMERIC CHARACTERS AND GRAPHIC DATA. THE DISPLAY REMAINS VISIBLE UNTIL IT IS ERASED AND DOES NOT REQUIRE CONTINUAL REFRESHING. THE DISPLAY AREA IS 14.5 BY 10.9 INCHES IN A 4096 BY 3120 GRID. FOUR CHARACTER SIZES ARE AVAILABLE. TO CHANGE CHARACTER SIZE, PUT THE LOCAL/LINE SWITCH ON LOCAL AND TYPE:

| | | | |
|----------------|---|------------------------------|----------|
| "ESC" KEY THEN | 8 | FOR MAXIMUM OF 74 CHARACTERS | 35 LINES |
| | 9 | 81 | 38 |
| | : | 121 | 58 |
| | ; | 133 | 64 |

THE : SIZE IS ADEQUATE FOR READING.

WHEN NO DATA IS INPUT FOR 30 SECONDS, THE SCREEN WILL DIM TO PROTECT THE PHOSPHOR. DEPRESS THE SHIFT KEY TO BRIGHTEN THE SCREEN AGAIN.

TO OPERATE AS AN INTERACTIVE TERMINAL TO CDC 6000 COMPUTERS, A 4800-BAUD ASCII, 1200-BAUD OR 300-BAUD MODEM IS REQUIRED.

- .TURN 4015 TERMINAL SWITCH AND HARD COPIER ON. SET LIGHT/DARK SWITCH ON HARD COPY UNIT FOR PROPER INTENSITY.
- .SET LOCAL/LINE SWITCH TO LINE.
- .WHILE TERMINAL AND HARD COPY UNIT WARM UP, CHECK THESE SWITCH SETTINGS:

- .ASCII/APL - ASCII
- .CODE EXPANDER - ON
- .CLEAR WRITE - OFF
- .ROTARY BAUD - EXTERNAL (REAR OF TERMINAL)
- .ASCII/BCD - ASCII (REAR OF TERMINAL)
- .RESET PAGE THEN DEPRESS SHIFT AND RESET PAGE KEYS TOGETHER.
- .DIAL PROPER PHONE FOR THE MODEM CONNECTED (SEE CCRM, 1-2)
- .WHEN COMPUTER DATA TONE ANSWERS, PUSH DATA BUTTON AND HANG UP.
- .LOG IN AND RUN JOB.
- .AT END OF RUN, IN COMMAND MODE, TYPE LOGOUT.
- .DISCONNECT THE PHONE LINE.

THE DTNSRDC KEYBOARD HAS APL CHARACTER SET ON TOP OF THE KEYS, SO THE PROPER CDC CHARACTER SET IS THE SMALL CHARACTERS ON THE FRONT OF SEVERAL OF THE SPECIAL SYMBOL KEYS. THE SIGNALS FOR SEVERAL CONTROL FUNCTIONS DIFFER FROM MOST INTERACTIVE TERMINALS:

| ACTION | OPERATION |
|---------------------------|-----------|
| ABORT THE CURRENT COMMAND | "BREAK" |
| BACKSPACE A CHARACTER | "RUBOUT" |
| CANCEL A LINE | CTRL-U |

WHEN USED AS AN INTERACTIVE TYPEWRITER TERMINAL, TWO COLUMNS OF COMMANDS WILL BE ACCUMULATED ON THE SCREEN UNLESS THE USER DEPRESSES THE "RESET PAGE" KEY. USER MUST RESET PAGE TO ALLOW NEXT SCREEN FULL OF DATA TO APPEAR WHEN LISTING OUTPUT. BEFORE USING GRAPHING ROUTINES ENTER "SCREEN(80,66)" COMMAND.

PLOTTING ROUTINES OF THE PLOT 10 PACKAGE ARE IN FILE "TEK48".

REFERENCES:

| | |
|--|-------------|
| PLOT-10 TERMINAL CONTROL SYSTEM USERS MANUAL | 062-1474-00 |
| PLOT-10 ADVANCED GRAPHING II USERS MANUAL | 062-1530-00 |

*** TEKTRONIX 4662 PLOTTER/DIGITIZER ***

THE TEKTRONIX 4662 PLOTTER/DIGITIZER MAY BE USED ON A 300-BAUD LINE WITH TERMINALS SIMILAR TO EXECUPORT 300, HAZELTINE 1000 (RS-232-C INTERFACE) AS A PEN PLOTTER USING THE TEKTRONIX PLOT 10 PACKAGE, OR AS AN ALPHAMERIC (95-CHARACTER ASCII) HARD COPY DEVICE. IT MAY ALSO BE USED AS A HARD COPY DEVICE ON A TEKTRONIX 4051.

SEVERAL SOFTWARE PACKAGES MAY BE USED WITH THE 4662. THE PLOT 10 PACKAGE IS LIBRARY "TEK30" CONTAINING 200 SUBROUTINES FOR USE WITH TEKTRONIX 401X STORAGE TUBE TERMINALS. SPECIAL SUBROUTINES SUCH AS "PLCN" AND "PLOFF", DIGITIZER CALL BUTTON AND OTHER SOFTWARE FOR THE SPECIAL FEATURES OF THE 4662 IS IN USER LIBRARY "TEK4662". REFERENCE IS "4662A01 PLOT 10 UTILITY ROUTINES FOR USE WITH THE 4662 PLOTTER USER MANUAL". MOST PROGRAMS WRITTEN FOR 11-INCH CALCOMP DRUM PLOTTERS MAY BE OUTPUT, WITH NO PROGRAM MODIFICATION, TO TEKTRONIX TERMINALS BY SUITABLE LIBRARY REPLACEMENT. BECAUSE THERE ARE DUPLICATE ENTRY POINTS IN THE ASSOCIATED LIBRARIES, THE LDSET MUST FOLLOW THE SPECIFIED ORDER OF LIBRARIES.

```
...
ATTACH,TEK30.
ATTACH,TEK4662.
ATTACH,CALC936.
LDSET,LIB=TEK4662/TEK30/CALC936.
```

COMPLETE PROGRAMS AVAILABLE INCLUDE:

```
"CAMVGCONTTSK,ID=CAMV", THE UNIVERSITY OF CALGARY GENERAL
PURPOSE CONTOURING PROGRAM (SEE "CONTOUR II: A REVISED
SURFACE FITTING AND MAPPING PROGRAM", CMD-76-26).
"CAMVPLT4662,ID=CAMV", A POSTPROCESSOR FOR THE "IMAGE" AND
"IDEAL" HARDCOPY FILES.
```

THE PLOTTER IS TURNED ON BY ROCKING THE POWER SWITCH TO THE RIGHT. BE SURE TO REMOVE THE PROTECTIVE CAP FROM THE SMALL FELT TIP PEN. THE 4662 CONTROL SWITCHES ARE NORMALLY SET TO "0221" (TERMINAL MUTE, RS-232 INTERFACE, DEVICE A, 300 BAUD). DEPRESS LOAD, PLACE ONE SHEET OF 11 BY 17 INCH PAPER ON THE PLATEN AND SMOOTH IT FLAT. PRESS LOAD AGAIN TO RELEASE BUTTON AND BUILD UP ELECTRO-STATIC CHARGE. WHEN LOAD BUTTON IS DOWN, NO OUTPUT ON THE TERMINAL SCOPE IS TRANSMITTED TO THE PLOTTER.

HARDWARE PAGE SCALE MARGINS MAY BE SET USING THE JOY STICK WHEN ONLY THE LOCAL BUTTON IS DEPRESSED. FIRST POSITION TO THE DESIRED LOWER LEFT CORNER AND PRESS THE SET LOWER LEFT BUTTON UNTIL IT BEEPS (1 SECOND), THEN POSITION TO THE DESIRED UPPER RIGHT CORNER AND SET IT SIMILARLY. THEN CHECK THE SETTINGS OF THE CURRENTLY DEFINED PAGE BY PRESSING IN SEQUENCE THE LOCATE BUTTONS. WHEN USED IN THE PLOT MODE, THE MARGINS MAY ALSO BE SET BY SOFTWARE IN THE PLOT-10 PACKAGE. IN LOCAL MODE THE KEYBOARD OF THE ATTACHED TERMINAL INPUTS ONLY TO THE PLOTTER, NOT OVER THE LINE TO THE COMPUTER.

JUNE 1977

PAGE 8-10

TO USE THE PLOTTER AS A HARD COPY ALPHAMERIC DEVICE FOR THE INTERACTIVE TERMINAL, THE LOCAL AND LOAD BUTTONS MUST BOTH BE UP. SIZE OF PRINTED LETTERS IS SCALED AUTOMATICALLY BY THE MARGIN SETTINGS TO ALLOW 85 CHARACTERS PER LINE AND 35 LINES. UNLESS THE SCREEN COMMAND IS USED, COMPUTER OUTPUT LINES WILL NOT EXCEED 72 CHARACTERS. CHARACTERS.

THE PLOTTER MAY BE USED AS A GIN (GRAPHIC INPUT) DIGITIZER DEVICE BY USING A SPECIAL CROSS HAIR MAGNIFIER INSTEAD OF THE BALL POINT PEN.

FOR FURTHER INFORMATION CONTACT MEL HAAS, CODE 1843, (202) 227-1933.

***** INTERCOM 4 *****

INTERCOM PERMITS SIMULTANEOUS REMOTE ACCESS TO THE CENTRAL COMPUTER RESOURCES IN AN INTERACTIVE MODE FOR USERS AT TELETYPE-COMPATIBLE OR MEDIUM SPEED REMOTE BATCH TERMINALS. INTERCOM OPERATES CONCURRENTLY WITH THE BATCH MODE ACTIVITY AT THE CENTRAL COMPUTER SITE AND GIVES TERMINAL USERS THE ABILITY TO:

- CREATE PROGRAMS BY ENTERING SOURCE STATEMENTS AT THE TERMINAL.
- CREATE, STORE, REFERENCE AND EDIT DISK FILES.
- SUBMIT FORTRAN EXTENDED, COBOL, MARS, ALGOL (THE SAME COMPILERS AS USED FOR REGULAR BATCH JOBS), OR BASIC PROGRAMS FOR COMPILATION AND EXECUTION UNDER INTERCOM CONTROL.
- CONVERSE INTERACTIVELY WITH EXECUTING PROGRAMS FROM THE TERMINAL.
- SUBMIT JOBS TO THE NOS/BE BATCH QUEUE FOR PROCESSING.
- ENTER NOS/BE CONTROL CARDS (UTILITIES) FOR PROCESSING.

INTERCOM PROVIDES FACILITIES WHICH HAVE BEEN PROVIDED IN THE PAST BY TIME SHARING SERVICES SUCH AS GE, MATRIX, CPS. THE COMMANDS AND FACILITIES ARE DESCRIBED IN DETAIL IN THE INTERCOM REFERENCE MANUALS (SEE CCRM, I).

*** SIGN-ON PROCEDURES ***

ALL INTERCOM USERS MUST REGISTER PASSWORDS (JOB ORDER NUMBERS) WITH CODE 1892 BEFORE THEY MAY HAVE ACCESS TO THE SYSTEM.

MOST INTERACTIVE TERMINALS ARE CONNECTED TO THE CENTRAL SITE BY DATA SETS. IF THE TERMINAL IS UNDER CENTREX (AT CARDEROCK, 227 IS CENTREX), DIAL 9 FIRST. MANY PORTS ARE AVAILABLE. IF THE COMPUTER IS DOWN DURING PRIME SHIFT, THE TERMINAL MAY AUTOMATICALLY TRANSFER TO THE RECORDED STATUS MESSAGE.

THE FOLLOWING PHONE NUMBERS ARE OF INTEREST:

| | |
|----------------|---|
| (301) 229-6000 | 6600 COMPUTER -- 300 AND 110 BAUD INTERACTIVE |
| (301) 229-6610 | 6600 COMPUTER -- 300 AND 110 BAUD INTERACTIVE |
| (301) 229-6622 | RECORDED COMPUTER STATUS MESSAGE |
| (202) 227-1907 | USER SERVICES FOR ASSISTANCE |

WHEN A TERMINAL IS CONNECTED, THE USER MUST ENTER CARRIAGE RETURN TO ESTABLISH THE BAUD RATE (300 OR 110). THEN THE SYSTEM RESPONDS WITH DATE AND TIME GREETING. THE USER BEGINS THE SESSION BY TYPING LOGIN. THE PERIOD IS OPTIONAL

IN RESPONSE TO ENTER USER NAME-
ENTER XXXXXXXXXX

WHERE XXXX IS THE USER'S INSTALLATION AND REGISTERED USER INITIALS, YYYYYY IS THE USER'S LAST NAME (OR FIRST SIX CHARACTERS OF IT).

IN RESPONSE TO M*****M ENTER PASSWORD-

ENTER, IN THE BLACKENED PASSWORD SPACE, THE REGISTERED PASSWORD TO WHICH THE SESSION IS TO BE CHARGED. ONLY PREVIOUSLY REGISTERED USER NAME/PASSWORD COMBINATIONS WILL BE ACCEPTED. IF A USER DOES WORK UNDER SEVERAL JOB NUMBERS, HE MUST REGISTER EACH WITH CODE 1892.

TO SPEED UP THE LOGIN PROCESS, A SHORT FORM IS AVAILABLE:
LOGIN,XXXXXXXX,SUP

IN ADDITION, A USER TURNKEY PASSWORD MAY BE REQUIRED (SEE NEXT PAGE).

*** USER TURNKEY PASSWORD ***

AN ADDITIONAL LEVEL OF SECURITY IS AVAILABLE THROUGH THE USE OF THE INTERCOM 'TURNKEY' COMMAND WHICH WILL DEFINE A 1- TO 9-CHARACTER USER TURNKEY PASSWORD WHICH MUST BE SPECIFIED AT FUTURE LOGINS UNTIL CHANGED OR REMOVED. THE TURNKEY IS A USER OPTION WHICH OFFERS PROTECTION AGAINST UNAUTHORIZED CHARGES TO THE USER'S INTERCOM ACCOUNT. IT SHOULD BE CHANGED FROM TIME TO TIME. THE INTERCOM TURNKEY DOES NOT AFFECT PERMANENT FILES OR BATCH USAGE.

WHEN A USER TURNKEY PASSWORD HAS BEEN DEFINED, LOGIN (BOTH LONG AND SHORT FORMS) WILL INCLUDE THE ADDITIONAL REQUEST:

***** ENTER TURNKEY PASSWORD

ENTER YOUR CURRENT TURNKEY IN THE BLACKENED SPACE.

THE TURNKEY COMMAND HAS THE FOLLOWING FORMS:

TURNKEY,NEW=NEWTKPASS

DEFINE A TURNKEY WHEN NONE CURRENTLY EXISTS.

TURNKEY,OLD=OLDTKPASS,NEW=NEWTKPASS

TURNKEY,NEW=NEWTKPASS,OLD=OLDTKPASS

REDEFINE AN EXISTING TURNKEY. BOTH OLD AND NEW TURNKEYS MUST BE GIVEN.

TURNKEY,OLD=OLDTKPASS,NEW=NULL

TURNKEY,NEW=NULL,OLD=OLDTKPASS

REMOVE A TURNKEY. THE OLD TURNKEY MUST BE GIVEN.

'OLDTKPASS' AND 'NEWTKPASS' ARE ANY COMBINATION OF 1-9 LETTERS AND/OR DIGITS (IF NO TURNKEY, USE 'NULL').

IF 'TURNKEY' IS SPECIFIED WITHOUT ARGUMENTS, THE USER WILL THEN BE PROMPTED FOR PASSWORDS AS FOLLOWS:

***** ENTER OLD TURNKEY PASSWORD

***** ENTER NEW TURNKEY PASSWORD

ENTER THE REQUESTED TURNKEYS IN THE BLACKENED SPACE.

USE OF THE TURNKEY IS OPTIONAL, BUT ONCE DEFINED, THE USER MUST REMEMBER IT, SINCE THE COMPUTER CENTER HAS NO RECORD OF IT. IF FORGOTTEN, THE INTERCOM ID MUST BE TOTALLY REPLACED BY USER SERVICES TO CLEAR THE TURNKEY, A CUMBERSOME PROCESS WHICH USUALLY REQUIRES 2 DAYS.

*** CORRECTING AND INTERRUPTING ***

1. TO DELETE A CHARACTER, TYPE CTRL BUTTON AND THE LETTER H SIMULTANEOUSLY, THEN RETYPE THE CHARACTER. (SEE NOTE 1)
2. TO CANCEL THE ENTIRE CURRENT LINE, HOLD THE CONTROL BUTTON WHILE TYPING X. RETYPE THE LINE CORRECTLY. (SEE NOTE 1)
3. TO EXIT FROM THE CURRENT COMMAND:
 - IF THE SYSTEM IS TYPING, HIT ESC KEY (OR ANY SINGLE CHARACTER), THEN PERCENT A, THEN RETURN
 - IF THE SYSTEM IS NOT TYPING, ENTER PERCENT A, THEN RETURN.AFTER USER ABORT, THE SYSTEM WILL REPLY COMMAND- OR .. FOR THE NEXT COMMAND.
4. HITTING THE ESC KEY (OR ANY SINGLE CHARACTER) FOLLOWED BY PERCENT S AND RETURN, WILL JUMP TO THE NEXT BUFFER, BUT WILL NOT ABORT EXECUTION OF THE CURRENT COMMAND.
5. TO DELETE A LINE FROM THE EDIT FILE, THE DELETE COMMAND MUST BE USED.
6. WHEN ANY EDITOR COMMAND GIVES LOADER ERROR, CHECK THAT IT WAS NOT INCORRECTLY TYPED WITH A PERIOD (.) AT THE END.
7. IF A USER IS ACCIDENTALLY DISCONNECTED FROM THE TTY (E.G., HIT EOT, CTRL-D, BY MISTAKE), HE WILL BE ABLE TO LOGIN WITHIN 15 MINUTES WITH ALL FILES PRESERVED. IF UNDER EDITOR, THE TEXT BUFFER WILL BE INTACT.
8. TO ENTER ANY COMMAND FROM KEYBOARD WHILE IN 'TAPE,ON' MODE, THE USER MUST END THE LINE BY CARRIAGE RETURN, LINE FEED, X-OFF (HOLD CTRL KEY WHILE TYPING LETTER S).
9. IF EXECUTION OF A COMMAND OR DIRECTIVE EXCEEDS 90 CPU SECONDS, THAT COMMAND IS ABORTED.
10. WHEN CURRENT EDITOR FILE HAS NOT BEEN SAVED, 'BYE' OR 'CREATE' OR 'EDIT' COMMANDS WILL NOT BE ACCEPTED UNTIL TYPED A SECOND TIME.
11. WHEN A PROGRAM IS ACCEPTING INPUT FROM THE TTY, SIMULATE END-OF-RECORD OR END-OF-FILE BY ENTERING PERCENT EOR OR PERCENT EOF, RESPECTIVELY. TO SIMULATE END-OF-INFORMATION, ENTER PERCENT EOF TWICE. THIS ALLOWS THE USER TO ENTER DATA FOR PROGRAMS (BOTH USER PROGRAMS AND SYSTEM PROGRAMS SUCH AS EDITLIB, UPDATE, ETC.) DIRECTLY AT THE TELETYPE. THIS FEATURE IS NOT THE SAME AS *EOR AND *EOF, WHICH ARE USED WHEN CREATING FILES IN EDITOR.
12. FILE NAMES BEGINNING WITH 'ZZZZZ' ARE RESERVED TO NOS/BE 1.0 AND SHOULD BE AVOIDED BY THE USER.

NOTE 1: THE CARRIAGE WILL NOT MOVE, NOR WILL A CHARACTER BE TYPED TO INDICATE YOUR CORRECTION. THE USER MUST KEEP TRACK OF WHAT HE IS DOING.

*** INTERCOM USER LIMITATIONS ***

REGISTRATION OF A JOB ORDER NUMBER FOR BATCH RUNS IS A PREREQUISITE FOR INTERCOM USAGE. IN ADDITION, EACH INDIVIDUAL MUST RECEIVE INTERCOM TRAINING AND PASSWORD ASSIGNMENT.

THE RUN COMPILER (FORTRAN) SHOULD NOT BE USED. ALL LIBRARY PROGRAMS ON THE SYSTEM FILE, EXCEPT THOSE IN THE FORTRAN MANUAL APPENDIX D, ARE CALLABLE ONLY BY FORTRAN EXTENDED (FTN).

CERTAIN COMPILERS NOMINALLY CALLABLE FROM INTERCOM REQUIRE FIXED OR MINIMUM FIELD LENGTHS LARGER THAN THOSE AVAILABLE TO GENERAL USERS. THESE INCLUDE MARS VI WITH UPDATE VERB, SNOBOL, FTN WITH DEBUG PACKETS.

MAXIMUM CPU TIME PER COMMAND OR PROGRAM EXECUTION IS 90 SECONDS. MAXIMUM CORE TO LOAD A USER PROGRAM FOR EXECUTION IS 61000 OCTAL WORDS. PERMISSION TO ALTER FIELD LENGTH UP TO 61000 OR TIME LIMIT UP TO 500 OCTAL SECONDS OF 6400 TIME IS GRANTED. (SEE CCRM, 9-15)

WHEN A FILE USED DURING A SESSION IS NO LONGER NEEDED, TYPE IN RETURN,LFN TO DELETE IT FROM USER LOCAL FILES. IT IS ESPECIALLY IMPORTANT TO RETURN PERMANENT FILES AS SOON AS POSSIBLE SINCE THERE IS A TOTAL SYSTEM MAXIMUM OF 200 ATTACHED PERMANENT FILES AT ANY ONE TIME. EXCEEDING THIS WILL HANG THE SYSTEM FOR ALL USERS.

USE OF THE ASSETS OR SUMMARY COMMANDS WHILE IN EDITOR WILL NOT INCLUDE THE CP TIME OF THE CURRENT EDITOR SESSION IN THE TOTALS.

ATTEMPT TO LOGIN WHEN INTERCOM IS NOT AVAILABLE OR SYSTEM IS HUNG RESULTS IN A DATA TONE THAT DROPS OUT OR TYPE HEAD CHATTERS WITH NO TYPING OR THE PHONE RINGS WITHOUT ANSWERING. FOR THE SYSTEM STATUS, CALL (301) 229-6622 (THIS IS A RECORDED MESSAGE).

WHEN THE SYSTEM HAS GONE DOWN WITHOUT NOTIFYING USER OR BEFORE USER COULD CATALOG CURRENT LOCAL FILES, TRY TO LOGIN AGAIN IN ABOUT 15 MINUTES. TYPE FILES COMMAND TO SEE IF THE USER LOCAL FILES HAVE BEEN RETAINED FROM PREVIOUS SESSION. IF SO, WORK MAY BE CONTINUED AND FILES CATALOGED.

AVOID THE TEACH COMMAND AS IT IS NOT EFFICIENT.

BINARY PROGRAMS BEING EXECUTED THROUGH INTERCOM MUST NOT BE ATTACHED WITH LOCAL FILE NAMES IDENTICAL TO ANY MEDIUM SPEED REMOTE BATCH TERMINAL COMMANDS. AVOID SINGLE LETTER NAMES, AS WELL AS THE COMMANDS THESE ABBREVIATE.

ALL FILES, INCLUDING THE SPECIAL FILES INPUT, OUTPUT, PUNCH AND PUNCHB, ARE RETURNED AUTOMATICALLY AT LOGOUT.

THE ONLY COMMAND WHICH AUTOMATICALLY CONNECTS INPUT AND/OR OUTPUT IS THE EDITOR RUN COMMAND. 'RUN,<COMPILER>' CONNECTS BOTH INPUT AND OUTPUT. 'RUN,<COMPILER>,NOGO' CONNECTS OUTPUT.

*** PAPER TAPE USE ***

TO USE PAPER TAPE FOR INPUT TO A RUNNING PROGRAM, EACH LINE TO BE READ MUST TERMINATE WITH CARRIAGE RETURN, LINE FEED, THREE RUBOUTS, AND X-OFF (HOLD CTRL KEY WHILE TYPING S), THEN THREE OR MORE RUBOUTS. IF A COMMAND TO INITIATE EXECUTION OF THE PROGRAM IS THE FIRST THING ON THE PAPER TAPE, THE TAPE MAY BE READIED BEFORE TYPING 'TAPE,ON'. OTHERWISE, LEAVE THE TAPE IN FREE OR STOP POSITION, REPLY TO THE .. BY TYPING THE INITIATE EXECUTION COMMAND (E.G. RUN,F) FOLLOWED BY CARRIAGE RETURN (-CR-), LINE FEED (-LF-), X-OFF. THEN PLACE THE TAPE IN AUTO MODE, OR WAIT FOR FIRST INPUT REQUEST, AND PUSH TAPE LEVER TO START.

TO READ PREVIOUSLY PREPARED PAPER TAPE UNDER EDITOR, READY THE TAPE IN THE READER, IN THE AUTO POSITION (JUST ABOVE STOP), TYPE 'TAPE,ON'. THE SYSTEM WILL RESPOND WITH .. AND START TO READ THE TAPE. ON MODEL 35 TTY, THE TD CALL-IN BUTTON MUST BE TURNED. IF THE TAPE DOES NOT START, USE THE START LEVER (TD ON). WHEN THE PAPER TAPE REACHES THE ENDING RUBOUTS, IT WILL STOP. IT MAY BE MANUALLY STOPPED BEFORE THAT OR MAY CONTAIN THE ENDING COMMANDS. TO EXIT FROM THE PAPER TAPE MODE, TYPE 'TAPE,OFF' FOLLOWED BY CARRIAGE RETURN, LINE FEED AND X-OFF.

WHEN IN THE TAPE,ON MODE ANY COMMAND TYPED IN ON THE CONSOLE MUST BE FOLLOWED BY CARRIAGE RETURN, LINE FEED, AND X-OFF BEFORE IT WILL BE ACCEPTED BY THE SYSTEM.

PAPER TAPE FOR EDITOR INPUT MAY BE PREPARED WITH LINE NUMBERS ON I.E., 100= ;PROGRAM SHORT(....) THE COMMAND SEQUENCE IS
EDITOR
D,A TO CLEAR EDIT FILE
TAPE,ON AND ALLOW THE TAPE TO READ IN IMMEDIATELY

WHEN PAPER TAPE FOR EDITOR INPUT DOES NOT CONTAIN ANY LINE NUMBERS (I.E., ;PROGRAM DIFF(....)) THE COMMAND SEQUENCE IS:
EDITOR
TAPE,ON PAPER TAPE MUST NOT BE READY IN THE AUTO POSITION.
C,S -CR-, -LF-, X-OFF FOR LINE COUNTER. THEN START THE PAPER TAPE
READING AFTER SYSTEM RESPONDS. TAPE WILL READ TO END
END SUPPLYING AUTOMATIC LINE NUMBERS.
= THIS CHARACTER ALONE ON A LINE OR AS LAST LINE ON PAPER
TAPE. IF TYPED IN, MUST HAVE -CR-, -LF-, X-OFF.
TAPE,OFF -CR-, -LF-, X-OFF
L,A LIST WHOLE PROGRAM TO CHECK SYSTEM ASSIGNED LINE
NUMBERS.

AFTER TERMINATING TAPE READ WITH 'TAPE,OFF', IF THE SYSTEM DOES NOT SEEM TO ACCEPT COMMANDS, THEN TYPE X-OFF AND REPEAT TAPE,OFF SEQUENCE.

*** THE ORDER OF LINE FEED AND CARRIAGE RETURN IN NOS/BE 1.0 IS THE SAME AS IN SCOPE 3.4 AND MUST BE THE REVERSE OF SCOPE 3.3 CUSTOM.

*** PROGRAM CONSTRUCTION - EDITOR ***

EDITOR MAY BE USED TO CREATE AND MODIFY PROGRAMS AND OTHER SEQUENTIAL FILES AT THE TELETYPE. ALL FILES SAVED UNDER EDITOR ARE ON PERMANENT FILE DEVICE. FILES OF GREATER THAN 72-CHARACTER RECORDS ARE ALLOWED. EACH LINE TO BE MANIPULATED BY EDITOR MUST HAVE A LINE NUMBER EITHER SUPPLIED BY THE USER OR GENERATED BY EDITOR. ANY SOURCE LANGUAGE WHICH SPECIFIES CERTAIN COLUMNS (E.G., FORTRAN STATEMENTS IN COLUMN 7) MUST EITHER BE SPACED OUT TO THE CORRECT COLUMN OR USE THE TAB CHARACTER (:). LENGTH OF LINES AND TAB STOPS ARE SET BY THE 'FORMAT' COMMAND.

ONLY THE LINES TO BE TYPED BY THE USER APPEAR BELOW.

FORTRAN

| | |
|--|--|
| EDITOR | ENTER EDITOR SUBSYSTEM |
| CREATE | SET AUTOMATIC LINE NUMBER GENERATION |
| :PROGRAM EXAMP(INPUT=128,OUTPUT=128,TAPES=INPUT) | |
| C AUTHOR AND ADDRESS | |
| C USES LIST-DIRECTED I/O | FOR COMMA SEPARATED UNFORMATTED DATA |
| :CALL CONNEC (5) | |
| :CALL CONNEC (6,OUTPUT) | |
| :PRINT *, "TYPE IN A,B,K " | |
| 2:READ(5,*) A,B,K | |
| :IF(A .EQ. 0.) GO TO 6 | |
| :C=A*B | |
| :PRINT 4,C,A,B,K | |
| 4:FORMAT(1X,3F7.2,I3) | |
| :GO TO 2 | |
| 6:STOP | |
| :END | |
| = | TURN OFF AUTOMATIC LINE NUMBER GENERATOR |
| LIST,ALL | LIST FOR PROOFREADING |
| SAVE,FNAME | MAKE USER LOCAL FILE |
| RUN,FTN | EXECUTION |

BASIC

| | |
|----------------------------|--------------------------------|
| EDITOR | |
| DELETE,ALL | TO CLEAR EDIT FILE |
| FORMAT,BASIC | ESTABLISH BASIC EDITING FORMAT |
| 100 REM COMPUTE AND PRINT | |
| 110 REM AUTHOR AND ADDRESS | |
| 120 LET P = 3.14159266 | |
| 130 PRINT " ENTER X"; | |
| 140 INPUT X | |
| 150 IF X<=0 THEN 200 | |
| 160 LET W = SQR(P * X) | |
| 170 PRINT " ROOT IS" W | |
| 180 GO TO 130 | |
| 200 END | |
| RUN,BASIC | EXECUTION |

CARD IMAGE FORTRAN DECK

| | |
|-------------------------------------|--------------------------------------|
| ATTACH,CARDS,ID=XXXX | BRING PERMANENT FILE TO USER FILES |
| EDITOR | |
| EDIT,CARDS,SEQ | SEQUENCE THE DECK FOR EDITING |
| L,A | LIST TO CHECK SEQUENCING; THEN EDIT |
| 90=:PROGRAM TEST(INPUT=128,OUT=128) | |
| /SIN/=COS/,A,U | TEXT REPLACEMENT, SIN INTO COS CALLS |
| /OUT/=OUTPUT/,90 | CORRECT FILE NAME IN PROGRAM CARD |
| D,210,220 | DELETE LINES |
| L,A | LIST FOR PROOFREADING |
| S,NEWPRO | MAKE USER LOCAL FILE |
| RUN,F | EXECUTE REVISED PROGRAM |
| STORE,NEWPRO,XXXX | CATALOG CORRECTED PROGRAM |

EXECUTION - EXTERNAL FILES REQUIRED

| | |
|------------------------------|---|
| ... | CREATION OR EDITING OF USER PROGRAM |
| RUN,F,N | COMPILE WITHOUT LOAD OR EXECUTE |
| BYE | MORE EFFICIENT TO EXIT EDITOR BEFORE COMMAND STRING |
| ATTACH,NSRDC. | GET NEEDED LIBRARY (PERIOD OPTIONAL) |
| XEC,LDSET=LIB=NSRDC,LOAD=LGO | |

BINARY PROGRAM RUNS

TO RUN A BINARY PROGRAM FILE INTERACTIVELY FROM INTERCOM, AFTER ATTACHING BE SURE THAT THE INPUT AND OUTPUT FILES ARE CONNECTED TO THE TERMINAL. EITHER USE

| | | |
|------------------------|-----------------|------------------|
| CONNECT,INPUT,OUTPUT | AS A COMMAND | OR |
| CALL CONNED (5LINPUT) | AND | (SEE FTN, 8-26; |
| CALL CONNED (6LOUTPUT) | IN FTN PROGRAM. | FTN/S, III-11-1) |

TO DETACH A FILE FROM THE TERMINAL, USE

| | | |
|-------------------|--------------------|----|
| DISCONT,LFN | AS A COMMAND | OR |
| CALL DISCON (LFN) | IN AN FTN PROGRAM. | |

*** BATCH RUNS FROM TELETYPE ***

TO RUN A JOB THAT REQUIRES MORE THAN 61000 CORE LOCATIONS, OVER 90 SECONDS, OR HAS MUCH OUTPUT, BATCH JOBS MAY BE INITIATED FROM A TTY. A COMPLETE CONTROL CARD RECORD MUST BE CONSTRUCTED UNDER EDITOR. SAVE THE FILE, ISSUE THE BATCH OR ROUTE COMMAND. TO RETURN JOB OUTPUT TO THE TTY, A DISPOSE OR ROUTE CARD MAY OCCUR IN THE CONTROL CARD RECORD, OR THE BATCH COMMAND MAY INCLUDE INPUT, HERE, OR THE ROUTE COMMAND MAY INCLUDE TID=XX, WHERE XX IS THE USER'S TERMINAL ID.

EXAMPLE: ROUTE,OUTPUT,DC=PR,TID=YB,FID=*. (SEE CCRM, 2-10) -OR-
DISPOSE,OUTPUT,PR=IYB. (SEE CCRM, 2-8)
SENDS ALL DATA CURRENTLY IN FILE OUTPUT TO THE TTY OR 200UT YB.

CREATION OF CONTROL CARD RECORD

EDITOR

C,S ENTER CREATE MODE WITH LINE NUMBERS SUPPRESSED
USERTE,CM60000,T500. NAME / CODE (YOUR JOB CARD)

CHARGE,USER,JJJJJJJJJJ,CC.

ATTACH,FNAME,ID=XXXX.

FTN,I=FNAME. WHERE FNAME IS THE FILE CONTAINING FTN SOURCE

LGO.

*FOR WILL GENERATE AN END-OF-RECORD (LIKE 7/8/9 CARD)
(YOUR INPUT DATA CARDS)

...

=

L,A

SAVE,SGRUN,N

BATCH,SGRUN,INPUT,HERE

TERMINATES CREATE MODE

LIST FOR PROOFREADING

SAVE CTL CDS WITHOUT LINE NUMBERS FOR BATCHING

OR ROUTE SGRUN,DC=IN,TID=XX, WHERE XX IS

YOUR TERMINAL ID. FOR CENTRAL SITE INPUT

QUEUE, OMIT ',HERE' FROM THE BATCH COMMAND

OR USE TID=C ON THE ROUTE COMMAND

TERMINAL ACCESS TO BATCH JOB OUTPUT

TO FOLLOW THE PROGRESS OF A SUBMITTED JOB, USE Q,XXXX (CCRM, 9-10) OR INTERCOM FILES COMMAND. THE Q COMMANDS DESCRIBED IN INT, II-1-36, SHOULD NOT GENERALLY BE USED AT INTERACTIVE TERMINALS.

WHEN THE REMOTE OUTPUT FILE MATCHING THE FIRST FIVE CHARACTERS OF YOUR JOB NAME APPEARS, THE JOB HAS BEEN EXECUTED. TO BRING THE REMOTE OUTPUT INTO A USER LOCAL FILE:

BATCH,CASGTNN,LOCAL

WHERE CASGTNN IS THE 7-CHARACTER REMOTE
JOB NAME

THE FILE IS NOW AVAILABLE FOR PAGE MANIPULATION AT THE TERMINAL
(SEE INT, II-1-21) OR EDITOR USE.

TO SEND OUTPUT TO A PRINTER AFTER TTY SAMPLING

REWIND,CASGTNN.

ROUTE,CASGTNN,DC=PR,TID=C,FID=*. TO CENTRAL SITE -OR-

ROUTE,CASGTNN,DC=PR,TID=AA,FID=*. WHERE AA IS 1700 OR 200UT REMOTE ID

*** LEAVING AND RE-ENTERING EDITOR ***

TO GET OUT OF EDITOR AND BACK INTO COMMAND MODE, ENTER BYE. IF THERE IS ANYTHING IN YOUR EDIT FILE, IT WILL BE RETAINED FOR NEXT ENTRY INTO EDITOR. THE CURRENT MODE (FORTRAN, BASIC, ETC.), CH, TABS AND TAB CHARACTER ARE ALSO SAVED, AS IS THE NUMBER OF THE LAST LINE ENTERED OR LISTED. THEN, IF YOU ENTER EDITOR AGAIN, THE MESSAGE YOU HAVE AN EXISTING EDIT FILE WILL BE TYPED AND YOUR LAST EDIT FILE AND ALL SETTINGS WILL BE AVAILABLE. NOTE THAT IT IS MORE EFFICIENT TO EXIT FROM EDITOR BEFORE DOING A STRING OF NOS/BE COMMANDS.

*** INTERCOM BACKUP DECKS ***

TO CREATE A BACKUP DECK FROM AN INTERCOM SOURCE CODE PERMANENT FILE THROUGH TELETYPE INPUT -

ATTACH,TEMP,ANY,ID=XXXX.
BATCH,TEMP,PUNCH,XXXX

GET YOUR FILE

THE SYSTEM WILL COPY THE PF AND PUNCH IT. DECK WILL HAVE BANNER CARD 'IXXXN' WHERE XXXX IS USER'S INITIALS. IF IT IS DESIRED TO PUNCH AT A 1700 INSTEAD OF CENTRAL SITE -

ATTACH,TEMP,PFN,ID=XXXX.
COPYE,TEMP,DISK

ROUTE,DISK,DC=PU,TID=AA,FID=*XXXX. SEE CCRM, 12-1 FOR AA VALUE

IN THIS CASE, THE BANNER CARD HAS 'XXXXNN'.

*** SIGN-OFF PROCEDURES ***

AT TERMINATION OF SESSION OR WHEN REQUESTED BY CENTRAL OPERATOR, TYPE 'LOGOUT.' (THE PERIOD IS OPTIONAL). USER MUST 'BYE' EDITOR BEFORE LOGOUT. A TIME SUMMARY OF THE SESSION AND ESTIMATED COST WILL BE PRINTED. WHEN THE SUMMARY HAS COMPLETED PRINTING, FINISH THE SIGN-OFF BY TURNING OFF THE TELETYPE AND HANGING UP THE DATA SET (PUSH TALK BUTTON) UNLESS ANOTHER USER IS WAITING TO LOGIN.

*** SYSTEM BULLETIN ***

A SYSTEM BULLETIN WILL BE TYPED AT LOGIN TIME UNLESS 'LOGIN,XXXXXXXXXX,JJJJJJJJJJ,SUP' WAS USED. SYSBULL MAY BE EXECUTED AT ANY POINT IN THE INTERCOM SESSION. 'CONNECT,OUTPUT' IS REQUIRED TO PRINT AT TTY ('SYSBULL,LOGIN' WILL ALWAYS PRINT AT TTY). (SEE CCRM, 2-7: SYSBULL)

*** LOCATING JOBS ***

Q COMMAND

THE 'Q' COMMAND LISTS JOBS IN THE VARIOUS QUEUES OF THE MAINFRAME ON WHICH THE USER IS EXECUTING. TO EXAMINE THE QUEUES OF OTHER MAINFRAMES, SEE THE 'QS' COMMAND ON THE NEXT PAGE.

- Q LIST NUMBER OF JOBS IN INPUT, OUTPUT, EXECUTE, PUNCH AND JANUS (CENTRAL SITE READ/PRINT/PUNCH) QUEUES.
- Q,JBN LIST NAMES OF JOBS IN INPUT, OUTPUT, EXECUTE, PUNCH AND AND JANUS QUEUES. JBN IS THE FIRST 3-7 CHARACTERS OF THE JOB NAME.
- Q,JBN,X LIST JOBS IN THE QUEUE SPECIFIED BY X WHOSE NAMES BEGIN WITH JBN. JBN IS 3-7 CHARACTERS AND X HAS ONE OF THE FOLLOWING VALUES:
I - INPUT QUEUE
O - OUTPUT QUEUE
E - IN EXECUTION
P - PUNCH QUEUE
J - JANUS QUEUE
APPROPRIATE STATISTICS ARE PROVIDED.
- Q,S SUMMARY OF HOW TO USE THE Q COMMAND.

NOTE: MOST 'Q' LISTS INCLUDE THE MAINFRAME (MFA=6700, MFB=6600, MFC=6400).

EXAMPLES

- Q,CASG LIST ALL JOBS SUBMITTED BY USER CASG
- Q,CASG,I LIST ALL JOBS SUBMITTED BY USER CASG WHICH ARE IN THE INPUT QUEUE
- Q,IPA LIST ALL ROUTED FILES OF USER PA
- Q,IPA,O LIST ALL ROUTED OUTPUT FILES OF USER PA

QS COMMAND

THE 'QS' COMMAND IS USED TO EXAMINE THE QUEUES ON THE USER'S AND OTHER MAINFRAMES. ALL PARAMETERS ARE OPTIONAL AND MAY BE GIVEN IN ANY ORDER.

QS, JOBNAME, FORMAT, TID, SID, TOTALS, QID

JOBNAME - FIRST 4-7 CHARACTERS OF DESIRED JOBNAME
(UP TO 5 JOBNAMES MAY BE SPECIFIED)
FORMAT - B OR OMITTED - BRIEF LISTING (DEFAULT)
V - VERBOSE LISTING
TID - TERMINAL ID TO BE EXAMINED
OMITTED - THE USER'S TERMINAL ID
AL - ALL TERMINAL ID'S
AA - WHERE AA IS ANY 2-CHARACTER TERMINAL ID
SID - SITE ID
MFA - 6700
MFB - 6600 (DEFAULT)
TOTALS - TOT - LIST QUEUE TOTALS
OMITTED - DO NOT LIST QUEUE TOTALS
QID - QUEUE TO BE EXAMINED
OMITTED - ALL QUEUES
I - INPUT QUEUE
T - TAPE JOBS IN INPUT QUEUE
O - OUTPUT QUEUE
E - EXECUTE QUEUE
P - PUNCH QUEUE
C - PLOT OUTPUT QUEUE

EXAMPLES

QS, TOT LIST MFB QUEUE TOTALS
(QS, TOT, AL WILL LIST ALL THE MFB QUQUES AND
THE TOTALS)

QS, TOT, MFA LIST MFA QUEUE TOTALS

QS, MFA LIST ALL MFA JOBS IN USER'S OWN QUEUES

QS, XXXX, AL, MFA LIST ALL MFA JOBS SUBMITTED BY USER XXXX

QS, XXXX, I, AL, MFA LIST ALL MFA JOBS SUBMITTED BY USER XXXX WHICH
ARE IN THE INPUT QUEUE

QS, IPAD LIST ALL MFB ROUTED FILES OF USER IPA AT THIS
TERMINAL ID
(NOTE: 4-CHARACTER MINIMUM JOB NAME)

QS, IPAD, XXXX, AL LIST ALL MFB JOBS FOR USER XXXX AND TERMINAL ID
PA

QS, MFA, T, V, PA, XXXX LONG (VERBOSE) LIST OF ALL TAPE JOBS FOR XXXX IN
MFA IN THE QUEUES FOR TERMINAL ID PA

*** MESSAGES BETWEEN CONSOLES ***

FROM CENTRAL SITE OPERATOR

THESE MESSAGES MAY NOT BE AVOIDED BY THE COMMAND LOCK,ON. THEY WILL PRINT IMMEDIATELY EVEN IF A PROGRAM IS IN EXECUTION, THEN RUN WILL RESUME. ALL MESSAGES BECOME PART OF THE DAYFILE RECORD. ALL MESSAGES FROM THE OPERATOR ARE PRECEDED BY THE TIME THE MESSAGE WAS SENT (SEE EXAMPLE AT BOTTOM OF PAGE).

CERTAIN FREQUENT MESSAGE PATTERNS ARE:

- ** , PLS LOGOUT GOING DOWN
DURATION OF INTERRUPTION USUALLY LONGER THAN 15 MINUTES.
OFTEN MAY NOT BE ABLE TO SAVE OR CATALOG FILES AS SYSTEM
MAY BE TOTALLY HUNG.
- ** , PLS LOGOUT IN XX MIN
- ** , PLS LOGOUT NOW
INTERCOM MUST BE DROPPED FOR A FEW MOMENTS. CATALOG FILES
AND LOGOUT AS SOON AS POSSIBLE. NOTE THAT THERE MAY NOT
BE ENOUGH TIME TO CATALOG BEFORE INTERCOM IS DROPPED.
- ** , INTERCOM OFF IN XX MIN
APPROACHING THE END OF THE SHIFT IN WHICH INTERCOM IS
AVAILABLE. BEGIN TO CATALOG FILES AND LOGOUT WITHIN XX
MINUTES.
- ** , PLS PURGE UNNEEDED FILES
- ** , PLS RETURN UNNEEDED FILES
DISK FILE SPACE IS FILLED. RETURN LOCAL FILES THAT ARE NO
LONGER REQUIRED AND/OR PURGE (AND RETURN) SOME FILES.
- ** , MACHINE IN STEP FOR 5 MIN
RESPONSE MAY BE VERY SLOW OR SEEM NONEXISTENT FOR A FEW
MINUTES. DEADSTART IS NOT YET PLANNED. BE PATIENT FOR 2
TO 10 MINUTES.

IF, INSTEAD OF **, THE MESSAGE CONTAINS A USER ID, IT IS NOT A BROADCAST MESSAGE, BUT IS SPECIFICALLY TO A TERMINAL USER. PLEASE ACKNOWLEDGE OR LOGOUT AT ONCE, IF REQUESTED.

14.26.57.PA, OK

RESPONSE TO USER PA AFTER A MESSAGE IS RECEIVED

TO CENTRAL SITE OPERATOR

M, REST OF MESSAGE UP TO 58 ALPHANUMERIC CHARACTERS.

DO NOT USE ANY SPECIAL CHARACTERS FROM PAGE 1-10 IN THE MESSAGE. M WILL SEND TO CENTRAL SITE FROM COMMAND OR EDITOR MODE ONLY IF NO OTHER MESSAGE IS PENDING AT THE CENTRAL CONSOLE FROM ANY USER. NO MESSAGES TO OPERATOR ARE STACKED. USER MUST RETYPE WHEN CONSOLE IS NOT BUSY. ALL RECEIVED MESSAGES BECOME PART OF THE SYSTEM DAYFILE.

THE CENTRAL SITE OPERATOR SHOULD NOT BE BOTHERED UNLESS A SYSTEM PROBLEM ARISES. FOR HELP WITH USE OF INTERCOM, EITHER TELEPHONE USER SERVICES OR TYPE SEND,USERSERVIC. (SEE BELOW)

TO ANOTHER TERMINAL

SEND COMMANDS MAY BE DIRECTED ONLY TO LOGGED IN INTERCOM TERMINALS (USE SITUATE TO VERIFY) WHO HAVE NOT TYPED LOCK,ON (USERS WHO HAVE TYPED LOCK,ON HAVE AN * PRECEDING THEIR NAMES IN THE SITUATE LIST). THE ENTIRE 8- TO 10-CHARACTER USER NAME MUST BE TYPED IN ANSWER TO WHOM (NOT THE 2-CHARACTER USER ID ALONE). IF TWO OR MORE USERS WITH THE SAME USER NAME ARE CURRENTLY LOGGED IN, PRECEDE THE USER NAME WITH THE 2-CHARACTER USER ID (E.G., SEND,PA-USERSERVIC). A CONVERSATION MAY BE CARRIED ON DURING THE SEND COMMAND AND MESSAGES WILL BE STACKED IN THE BUFFER OF THE RECEIVER UNLESS LOCKED. ALL MESSAGES ARE TIME STAMPED (LEVEL 420). IT IS NOT NECESSARY TO TYPE END TO RECEIVE A REPLY TO FIRST PART OF A MESSAGE. TO TERMINATE THE SEND, END MUST NOT BE FOLLOWED BY ANY OTHER CHARACTERS, INCLUDING PERIOD. AVOID SEND TO *** UNLESS A SYSTEM PROBLEM IS INVOLVED.

EXAMPLE: SEND,USERSERVIC
PLEASE CALL 555-1234 TO HELP USER
EDITOR PROBLEMS WITH PERM FILE
END

SEND, AS DESCRIBED ABOVE, WILL PROMPT EACH USER ENTRY WITH TYPE MESSAGE OR END-. TO SUPPRESS THIS PROMPTING, USE SEND,XXXXXXXXXX,S, WHICH WILL GIVE AN INITIAL PROMPT OF GO- AND THEN ACCEPT AND SEND USER MESSAGES UNTIL END IS TYPED.

TO A USER WHO IS NOT LOGGED IN

MESSAGES MAY BE SENT TO INTERCOM USERS WHO ARE NOT LOGGED IN BY USING BEGIN,SEND. IN ORDER TO RECEIVE MESSAGES, A USER MUST FIRST HAVE A PERMANENT FILE NAMED 'SENDFIL'. THE SEND PROCEDURE HAS 4 FUNCTIONS:

| | |
|-------------------------|--|
| BEGIN,SEND,,CREATE | CREATE PERMANENT FILE 'SENDFIL' |
| BEGIN,SEND,,ANY | CHECK TO SEE IF THERE ARE ANY MESSAGES |
| BEGIN,SEND,,LIST,OUTPUT | LIST MESSAGES ON FILE 'OUTPUT' |
| BEGIN,SEND,,SEND,TO | SEND MESSAGES TO USER 'TO' (4-CHARACTER USER-ID). MAY BE ABBREVIATED 'BEGIN,SEND'. |
| BEGIN,SEND,,EXPLAIN | TYPE EXPLANATION OF SEND PROCEDURE |

*** SAVING FILES ***

WHEN A USER DESIRES TO KEEP A PROGRAM OR DATA FILE FOR USE AT ANOTHER SESSION, IT MUST BE CATALOGED INTO THE PERMANENT FILE SYSTEM.

SAVE,LFN THIS WILL BE ON A PF DEVICE (EDITOR)
CATALOG,LFN,PFN,ID=XXXX,AC=JJJJJJJJJJ,<PARAMETERS>.
ENTER FILE IN CATALOG (AC OPTIONAL)

SEE ALSO 'STORE' COMMAND BELOW.

*** STORE, FETCH, DISCARD COMMANDS ***

INTERCOM COMMANDS STORE, FETCH, AND DISCARD HAVE BEEN MODIFIED TO BE COMPATIBLE WITH DTNSRDC ACCOUNT CONVENTIONS. THE CURRENT PASSWORD JOB ORDER NUMBER WILL BE THE AC TO CATALOG THE FILE. IF THE XXXX 4-CHARACTER ID FOR THE FILE IS OMITTED, IT IS TAKEN FROM THE LOGIN USER ID.

STORE,LFN
STORE,LFN,XXXX IS THE EQUIVALENT OF:
 CATALOG,LFN,ID=XXXX,AC=JJJJJJJJJJ,CY=999.
 WHERE JJJJJJJJJJ IS THE JOB ORDER NUMBER FROM
 'LOGIN,XXXX,JJJJJJJJJJ'.
 THE STORE COMMAND WILL PUT THE FILE ON A *PF DEVICE
 IF IT IS NOT INITIALLY. AS WITH A CATALOG, A 2-LINE
 MESSAGE WILL BE TYPED IF THE CATALOG IS SUCCESSFUL.
 THE STORE COMMAND HAS NO PROVISION FOR PASSWORDS.

FETCH,LFN
FETCH,LFN,XXXX IS THE EQUIVALENT OF:
 ATTACH,LFN,ID=XXXX.
 NO MESSAGE WILL BE TYPED IF THE ATTACH IS SUCCESSFUL.

DISCARD,LFN
DISCARD,LFN,XXXX IS THE EQUIVALENT OF:
 ATTACH,LFN,ID=XXXX.
 PURGE,LFN.
 RETURN,LFN.
 IF LFN WAS SUCCESSFULLY PURGED, A 2-LINE MESSAGE
 WILL BE TYPED.

*** ADDITIONAL INTERCOM COMMANDS ***

DAY,OFF SUPPRESS PRINTING OF DAYFILE MESSAGES AT THE TERMINAL.

DAY,CN PRINT DAYFILE MESSAGES AT THE TERMINAL. (DEFAULT)

EFL,XXXXX CHANGE EXECUTION FIELD LENGTH TO XXXXX (NORMALLY UP TO 61000). THE SECOND AND THIRD FORMS ARE USED TO RESTORE THE DEFAULT FIELD LENGTH (46000).

EFL,0

EFL

ETL,XXX CHANGE THE COMMAND TIME LIMIT TO XXX OCTAL SECONDS (NORMALLY UP TO 500 (320 DECIMAL)). THE SECOND AND THIRD FORMS ARE USED TO RESTORE THE DEFAULT COMMAND TIME LIMIT OF 130 (90 DECIMAL) SECONDS. A SHORT TIME LIMIT IS USEFUL IN DEBUGGING A LOOPING PROGRAM (MINIMIZES CHARGES). WHEN THE TIME LIMIT HAS BEEN REACHED, THE MESSAGE 'CP TIME LIMIT' WILL BE TYPED.

ETL,0

ETL

SCREEN,XXX,YY WHEN A PROGRAM PRINTS OUTPUT ON AN INTERCOM TERMINAL, THE SYSTEM GENERATES A 'GO TO NEXT LINE' AFTER EACH 72 CHARACTERS EVEN IF THE OUTPUT IS LONGER. ON WIDE CARRIAGE TERMINALS, IT IS POSSIBLE TO PRINT UP TO 132 CHARACTERS PER LINE BY TYPING 'SCREEN,132'. THE NUMBER OF CHARACTERS PER LINE IS SWITCHED TO 132. ANY NUMBER LESS THAN 132 MAY BE USED, SUCH AS 'SCREEN,120'. TO RESET FROM LONG LINES BACK TO DEFAULT, TYPE 'SCREEN'. A USER AT A 72-CHARACTER/LINE TERMINAL DEVICE SHOULD NOT USE THE COMMAND SINCE ALL CHARACTERS AFTER 72 WOULD OVERPRINT THE LAST CHARACTER AT THE END OF THE LINE. 'YY' SPECIFIES THE NUMBER OF VERTICAL LINES ON A DISPLAY TERMINAL. IT IS IGNORED FOR TELETYPES AND SIMILAR TERMINALS. (SEE CCRM, 10-18: ITEM 8, FOR AN EXAMPLE OF SCREEN,XXX,YY.)

SCREEN,XXX

SCREEN

*** XEQ COMMAND ***

THE INTERCOM 'XEQ' COMMAND MAY BE USED TO EXECUTE PROGRAMS. IT IS REQUIRED TO MAKE USE OF CERTAIN LOADER COMMANDS: EXECUTE, LDSET, LIPLCAD, LOAD, NOGO, SLOAD, SATISFY. (SEE CCRM, 2-15,16)

TO INITIATE PROGRAM LOADING, ENTER:

XEQ

THE SYSTEM RESPONDS:

OPTION=

ENTER ONE OF THE OPTIONS (LOADER COMMANDS). IF EXECUTE, NOGO OR A FILENAME IS NOT ENTERED, THE SYSTEM CONTINUES TO REQUEST OPTION= ENTRIES. TO LEAVE THE XEQ COMMAND, RESPOND TO THE OPTION= REQUEST WITH 'END'.

IF THE LOAD SEQUENCE ENDS WITH EXECUTE, NOGO OR A FILENAME, THE SHORT FORM:

XEQ,OPTION1,OPTION2,....,OPTIONN

MAY BE USED.

WHEN EXECUTING A PROGRAM BY FILENAME (E.G., 'LGO'), EITHER 'LGO' OR 'XEQ,LGO' OR 'XEQ,LOAD=LGO' MAY BE USED.

OPTIONS= ENTRIES ARE ENTERED AS FOLLOWS:

EXECUTE=ENAME,PARAM1,PARAM2,....,PARAMN

EXECUTE=,PARAM1,PARAM2,....,PARAMN

NOGO=NAME

NOGO

NAME,PARAM1,PARAM2,....,PARAMN

LOAD=LFN1,LFN2,....,LFNN

LDSET,OPTION1,OPTION2,....,OPTIONN

LDSET=OPTION1,OPTION2,....,OPTIONN

FILES=LFN1/LFN2/.../LFNN

FOR ADDITIONAL INFORMATION ON THE VARIOUS OPTIONS, SEE CCRM, 2-15,16; INTERCOM, II-1-38 THRU 42.

XEQ EXAMPLES

1. PROGRAM USES SUBROUTINES ON LIBRARY 'NSRDC'
ATTACH,NSRDC.
FTN,I=PROG,T,L=OUT.
XEQ,LDSET,LIB=NSRDC,LGO
2. MAIN PROGRAM USES ALL SUBROUTINES ON FILES 'SUBS' AND 'MORSUBS'.
MAIN PROGRAM IS IN EDITOR WORKING STORAGE.
RUN,FTN,N
XEQ,LOAD=LGO,SUBS,MORSUBS,EXECUTE
3. USER WANTS TO PRESET CORE TO NEGATIVE INFINITE WITH EACH WORD
CONTAINING ITS OWN ADDRESS. A FULL MAP WITH CROSS REFERENCE IS
DESIRED FOR OFFLINE PRINTING. PROGRAM IS IN FILE 'LGO'.
DISCONT,OUTPUT
XEQ,LDSET=MAP=SBX,PRESETA=NGINF,LOAD=LGO
DISPOSE,OUTPUT,PR=C.
4. SAME AS 3. ONLY OVERRIDE FIRST PROGRAM CARD FILE NAME WITH FILE NAME
'TAPE14'.
DISCONT,OUTPUT
XEQ,LDSET,MAP=SBX,PRESETA=NGINF,LOAD=LGO,EXECUTE=,TAPE14
5. CREATE A CORE IMAGE MODULE ('CIM') FROM THE RELOCATABLE MODULE(S)
ON FILE 'LGO'
XEQ,LOAD=LGO,NOGO=CIM

THE CORE IMAGE MODULE 'CIM' MAY NOW BE USED TO TEST THE PROGRAM
USING SEVERAL SETS OF TEST DATA BY USING A COMMAND SUCH AS ONE OF
THE FOLLOWING:

CIM.
CIM,TEST1.
CIM,TEST2.

WHERE 'TEST1' AND 'TEST2' ARE FILES CONTAINING TEST DATA AND
OVERRIDE THE FIRST PROGRAM CARD FILE NAME. THIS WILL SAVE LENGTHY
LOADER PROCESSING SINCE THE LOADER WILL NOT HAVE TO FIND ALL THE
REQUIRED SUBPROGRAMS FOR EACH LOAD. IF THE PROGRAM IS FOUND TO
WORK, 'CIM' CAN THEN BE CATALOGED (PROVIDED A REQUEST,CIM,*PF.
PRECEDED THE XEQ).

PAGE 9-18

LOCAL FILES MAY BE EXAMINED USING THE PAGE COMMAND (SEE INT, II-1-21). THE FILE MAY BE POSITIONED FORWARD OR BACKWARD; TABS ARE PROVIDED FOR SCANNING LONG LINES; THE PRESENCE OF ABSENCE OF A CHARACTER STRING MAY BE TESTED; SELECTED LINES MAY BE TRANSFERRED TO A FILE FOR LATER PRINTING; PAGE MAY BE EXITED AND RE-ENTERED WITHOUT LOSS OF PARAMETER INFORMATION OR THE LINE INDEX FILE. PAGE IS GENERALLY FASTER THAN EDITOR FOR LOOKING AT AN OUTPUT FILE.

THE FOLLOWING CONTROL CARD WILL INITIALIZE THE PAGE PROGRAM:

PAGE WILL RESPOND WITH:

AFTER WHICH ANY NUMBER OF PAGE COMMANDS ARE ENTERED IN ONE LINE, SEPARATED BY COMMAS. THESE COMMANDS ARE EXECUTED CONSECUTIVELY. IF MORE THAN ONE DISPLAY COMMAND IS GIVEN, ONLY THE FINAL DISPLAY-PAGE IS TYPED.

IF AN ERROR IS FOUND IN A COMMAND (SEQUENCE), THE CURRENT
DISPLAY-PAGE IS TYPED, FOLLOWED BY U/XXX... , WHERE XXX... IS THE
UNPROCESSED PORTION OF THE COMMAND (SEQUENCE).

HELP COMMAND

PAGE INCLUDES SEVERAL LISTS WHICH EXPLAIN THE VARIOUS COMMANDS.
ENTER HELP FOR A DESCRIPTION OF THE LISTS AVAILABLE.

GENERAL CONTROL COMMANDS

R REPEAT THE PREVIOUS COMMAND LINE
E OR Q EXIT THE PAGE PROGRAM

DISPLAY FORMAT COMMANDS

TNN DISPLAY LINE BEGINNING AT CHARACTER POSITION NN
(INITIALLY SET TO T1)
S DISPLAY THE FIRST 72 CHARACTERS OF THE LINE.
(S MODE IS DEFAULT)
TO SEE MORE OF THE LINE, ENTER T73. TO RESTORE TO THE
FIRST 72 CHARACTERS, ENTER T1.
F DISPLAY EACH NON-BLANK LINE IN ITS ENTIRETY, USING SEVERAL
DISPLAY LINES IF NECESSARY.

LINE LOCATION AND PAGE SEARCHING COMMANDS

NN GO TO LINE NN
(DISPLAY 10 LINES STARTING AT LINE NN)
+NN GO FORWARD NN LINES
(DISPLAY 10 LINES BEGINNING NN LINES AFTER START OF LAST
DISPLAY-PAGE)
-NN GO BACK NN LINES
(DISPLAY 10 LINES BEGINNING NN LINES BEFORE START OF LAST
DISPLAY-PAGE)
+ GO FORWARD ONE DISPLAY-PAGE
(DISPLAY NEXT 10 LINES)
- GO BACK ONE DISPLAY-PAGE
(DISPLAY PREVIOUS 10 LINES)
* GO FORWARD TO END-OF-FILE
H OR +H GO FORWARD TO NEXT HEADER LINE
(A HEADER LINE HAS A '1' IN COLUMN ONE)
-H GO BACK TO LAST HEADER LINE

JUNE 1977

PAGE 9-20

STRING SEARCHING COMMANDS

THESE COMMANDS SEARCH FORWARD OR BACKWARD FOR THE PRESENCE OR ABSENCE OF A SPECIFIED CHARACTER STRING. THE FIRST LINE SATISFYING THE SEARCH CONDITION IS DISPLAYED AS THE FIRST LINE OF A PAGE OF INFORMATION. IF A SEARCH CONDITION IS NOT MET, A MESSAGE STATES THAT THE BEGINNING OR END OF THE FILE HAS BEEN REACHED.

STRING SEARCH COMMANDS HAVE THE FOLLOWING FORM:

| DIRECTION | TYPE | RANGE | CONDITION | STRING |
|--------------|---|-------|-----------|--------|
| WHERE | | | | |
| DIRECTION | | | | |
| + OR OMITTED | SEARCH FORWARD | | | |
| - | SEARCH BACKWARD | | | |
| TYPE | | | | |
| OMITTED | SEARCH EVERY LINE | | | |
| H | SEARCH HEADER LINES | | | |
| RANGE | | | | |
| OMITTED | STRING IS ANYWHERE IN THE LINE | | | |
| (I) | STRING MUST START IN COLUMN I | | | |
| (I-J) | STRING MUST START SOMEWHERE BETWEEN COLUMNS I AND J, INCLUSIVE | | | |
| CONDITION | | | | |
| = | STRING MUST BE IN THE LINE | | | |
| " OR ≠ | STRING MUST NOT BE IN THE LINE | | | |
| STRING | | | | |
| DXXX...D | D IS THE DELIMITER AND IS ANY CHARACTER NOT IN THE SEARCH STRING XXX... | | | |

EXAMPLES OF STRING SEARCHING COMMANDS:

| | |
|------------------|--|
| =/ABCD/ | SEARCH FORWARD FOR THE FIRST OCCURRENCE OF ABCD ANYWHERE IN A LINE AND LIST 10 LINES |
| D,H"/FTN/ | GO TO BEGINNING OF FILE, THEN SEARCH FORWARD FOR THE FIRST HEADER LINE WHICH DOES NOT CONTAIN FTN. (0, IS NOT NEEDED IF ALREADY AT START OF FILE.) |
| -(12)=*SEVERITY* | SEARCH BACKWARD FOR THE FIRST LINE WHICH HAS SEVERITY STARTING IN COLUMN 12. THIS CAN BE USED TO LOCATE FTN ERROR DIAGNOSTICS IN A COMPILATION LISTING. TO SEARCH FORWARD, REMOVE THE MINUS (-). |
| -H(1-9)=*VER* | SEARCH BACKWARD FOR FIRST HEADER LINE WHICH HAS VER STARTING SOMEWHERE IN COLUMNS 1-9, INCLUSIVE. |

JUNE 1977

PAGE 9-21

*** TEXT EDITOR ***
(NETED)

AN ELEMENTARY TEXT EDITOR, CALLED 'NETED', IS NOW AVAILABLE ON THE DTNSRDC CDC 6000 COMPUTERS. IT HAS SOME FEATURES NOT AVAILABLE IN THE INTERCOM EDITOR. NETED USES VERY LITTLE CORE (CURRENTLY 13K) AND LESS CP TIME THAN EDITOR.

NETED OBTAINED FROM THE LAWRENCE BERKELEY LABORATORY (BKY) AND WAS WAS ENHANCED BY ONE OF OUR CUSTOMERS (THE CONSTRUCTION ENGINEERING RESEARCH LABORATORY (CERL), COMPUTER SERVICES BRANCH).

A DOCUMENT (22 PAGES+COVER) DESCRIBING NETED MAY BE OBTAINED BY:

BEGIN,UTILITY,,PROGDOC,OTHER,,NETED,OUTPUT.

JUNE 1977

PAGE 10-1

***** LIBRARY AND OTHER ROUTINES *****

SYSTEM OBJECT ROUTINES NEEDED BY COMPILED PROGRAMS ARE AVAILABLE IN SYSTEM LIBRARIES AND ARE DISCUSSED BELOW. ADDITIONAL OBJECT ROUTINES HAVE BEEN ADDED IN LIBRARIES OF SUBPROGRAMS (10-2), LIBRARIES OF MAIN PROGRAMS (10-19) AND SOME MISCELLANEOUS PROGRAMS (10-21). SOME OF THESE ROUTINES ARE FROM VIM (THE CDC 6000 USER'S GROUP), FROM OTHER ORGANIZATIONS AND SOME HAVE BEEN WRITTEN AT DTNSRDC. THESE ARE DISCUSSED IN THE REST OF THE CHAPTER.

*** NOS/BE 1.0 SYSTEM LIBRARIES ***

NOS/BE 1.0 HAS SEVERAL LIBRARIES. NO ATTACH IS USED FOR ANY OF THE SYSTEM FILES.

THE COMPILERS GENERATE, AS PART OF THEIR OUTPUT OBJECT CODE, THE REQUIRED LDSET FOR THE NORMAL OBJECT LIBRARY. IF A COBOL PROGRAM CALLS FTN OBJECT LIBRARY ROUTINES OR AN FTN PROGRAM CALLS COBOL OBJECT LIBRARY ROUTINES, THE USER MUST SUPPLY THE ADDITIONAL LDSET.

| LIBRARY | CONTENTS |
|---------|--|
| BASLIB | RELOCATABLE ROUTINES FOR BASIC |
| COBOL | COBOL 4.6 OBJECT ROUTINES |
| FORTAN | FTN 4.6 OBJECT ROUTINES (FTN, CHAPTERS 8, 9, I/O; FTN/S, I-8-2, I-8-7, I-8-9, III-11-1, I/O; DEBUG) |
| IGS274 | 274 IGS 2.0 OBJECT LIBRARY |
| NUCLEUS | STANDARD COMPILERS (E.G., COBOL, FTN) SYSTEM CONTROL CARD UTILITIES (E.G., COPYCF, UPDATE, AUDIT) INTERCOM COMMANDS (E.G., BATCH, LOGIN, XEQ) |
| RUN2P3 | RUN FORTRAN 2.3 OBJECT ROUTINES |
| SIMLIB | RELOCATABLE ROUTINES FOR SIMSCRIPT I.5 |
| SRTLIR | RELOCATABLE SORT/MERGE ROUTINES USED BY COBOL AND FTN |
| SYSIO | RECORD MANAGER ROUTINES, INTERNAL I/O MACROS, CPC ROUTINES, FORM OBJECT ROUTINES AND SYMPL OBJECT LIBRARY |
| SYSOVL | HIGHER LEVEL OVERLAYS OF COMPILERS IN NUCLEUS |

*** LIBRARIES OF SUBPROGRAMS ***

LIBRARY FILES EXIST FOR EACH MAJOR PACKAGE OF SUBPROGRAMS. EACH WILL REQUIRE ATTACH AND LDSET/LIBRARY CARDS TO INITIATE AUTOMATIC SUBROUTINE LOADING. PROGRAMS USING ROUTINES FROM ANY OF THESE LIBRARIES MUST HAVE BEEN COMPILED USING NOS/BE 1.0 OR SCOPE 3.4.

TO ACCESS ANY OF THESE PUBLIC LIBRARIES:

| | |
|-------------------|--|
| ATTACH,LIBNAM. | WHERE LIBNAM IS A LIBRARY DESCRIBED BELOW |
| . | |
| . | PREPARE USER OBJECT PROGRAM |
| . | |
| LDSET(LIB=LIBNAM) | OR LDSET(LIB=LIBNAM1/LIBNAM2) IF SEVERAL |
| LGO. | LOAD USER PROGRAM WHICH CALLS ROUTINES ON THE SPECIFIED LIBRARY |

SCIENTIFIC LIBRARIES CURRENTLY INCLUDE: ARLNALG, EDSTAT, EISPACK,
FUNPACK, IMSL, MSL, NSRDC, SSP

PLOTTING LIBRARIES CURRENTLY INCLUDE: CALCFN, CALC170, CALC3D, CALC763,
CALC936, SCCALC, SC4020, SC4060

PRINTER PLOTS ARE IN: NSRDC, SSP

MOST OF THE LIBRARIES ARE NON-PROPRIETARY AND HAVE BOTH SOURCE AND OBJECT CODE AVAILABLE TO THE USER. A FEW ARE PROPRIETARY AND HAVE ONLY THE OBJECT CODE AVAILABLE.

SYSMISC

PERMANENT FILE 'SYSMISC' CONTAINS THE OBJECT ROUTINES FOR FTN 3.0, COBCL 3.0, SORT 3.0, AND SIMULA. TO USE THESE ROUTINES, THE JOB MUST ATTACH THE LIBRARY PRIOR TO LOADING THE RELOCATABLE PROGRAM:

| | |
|--------------------------|---------------------------------|
| ... | |
| ATTACH,SYSMISC. | |
| ... | |
| LDSET,LIB=SYSMISC/SYSIO. | |
| LGO. | ** LOAD OLD RELOCATABLE PROGRAM |

ARLNALG (NON-PROPRIETARY)

THE AEROSPACE RESEARCH LABORATORIES (ARL) LINEAR ALGEBRA LIBRARY IS A COLLECTION OF 34 SUBROUTINES FOR SOLUTIONS TO LINEAR SYSTEMS AND DETERMINATION OF EIGENVALUES AND EIGENVECTORS OF REAL SYMMETRIC MATRICES. SOME OF THESE ROUTINES ARE SPECIFICALLY OPTIMIZED FOR 6000. (SEE CCLIB AND THE ARL LINEAR ALGEBRA LIBRARY HANDBOOK (TR 74-0106))

EDSTAT (PROPRIETARY)

LIBRARY EDSTAT CONTAINS BOTH THE EDSTAT-P AND EDSTAT-V ROUTINES FOR STATISTICAL ANALYSIS OF BEHAVIORAL SCIENCE DATA WHICH WERE OBTAINED FROM DATA SCIENCES ASSOCIATES.

EISPACK (NON-PROPRIETARY)

THE EIGENSYSTEM PACKAGE FROM ARGONNE NATIONAL LABORATORY IS A COLLECTION OF OVER 35 SUBROUTINES TO SOLVE EIGENVECTOR AND EIGENVALUE PROBLEMS. ROUTINES IN THIS PACKAGE ARE OFTEN SUPERIOR IN SPEED AND ACCURACY TO SIMILAR ROUTINES IN OTHER PACKAGES. (SEE CCLIB)

MACHINE-READABLE DOCUMENTATION MAY BE LISTED USING
JOBNAME,MT1. NAME / CODE
CHARGE,.....
BEGIN,DOCTAPE,,EISPACK,<ROUTINE>,OUTPUT.
(OUTPUT MAY BE ROUTED TO PRINT ON NARROW PAPER)

REFERENCE: MATRIX EIGENSYSTEM ROUTINES - EISPACK GUIDE; SMITH,
BOYLE, GARBOW, IKEDE, KLEMA, MOLER.

FUNPACK (NON-PROPRIETARY)

SPECIAL FUNCTIONAL SUBROUTINE PACKAGE FROM ARGONNE NATIONAL LABORATORY CONTAINING 32 ROUTINES FOR BESSEL FUNCTIONS, DAWSON'S INTEGRAL, ELLIPTIC INTEGRALS OF THE FIRST AND SECOND KIND AND EXPONENTIAL INTEGRAL. (SEE CCLIB)

MACHINE-READABLE DOCUMENTATION MAY BE LISTED USING
JOBNAME,MT1. NAME / CODE
CHARGE,.....
BEGIN,DOCTAPE,,FUNPACK,<ROUTINE>,OUTPUT.
(OUTPUT MAY BE ROUTED TO PRINT ON NARROW PAPER)

IMSL (PROPRIETARY)

THE INTERNATIONAL MATHEMATICAL AND STATISTICAL LIBRARIES PACKAGE CONTAINS OVER 300 SUBROUTINES IN THE FOLLOWING AREAS:

- .ANALYSIS OF EXPERIMENTAL DESIGN DATA
- .RANDOM NUMBERS, GENERATION AND TESTING
- .STATISTICS, BASIC, NON-PARAMETRIC, SPECIAL FUNCTIONS
- .REGRESSION ANALYSIS
- .DIFFERENTIAL EQUATIONS, INTERPOLATION, APPROXIMATION, SMOOTHING
- .LINEAR ALGEBRAIC EQUATIONS
- .VECTOR MATRIX ARITHMETIC

EDITION 5 OF IMSL CONTAINS ALL PREVIOUS UPDATES AND INCLUDES OVER 50 NEW SUBROUTINES.

MACHINE-READABLE DOCUMENTATION MAY BE LISTED USING

JOBNAME, MT1.

NAME / CODE

CHARGE,.....

BEGIN, DOCTAPE,, IMSL, <ROUTINE>, OUTPUT.

(OUTPUT MAY BE ROUTED TO PRINT ON NARROW PAPER)

MSL (PROPRIETARY)

THE CDC MATH SCIENCE LIBRARY (BOEING PACKAGE) CONTAINS OVER 200 NUMERICAL MATHEMATICAL ROUTINES COVERING THE FOLLOWING EIGHT AREAS:

- .PROGRAMMED ARITHMETIC
- .ELEMENTARY FUNCTIONS
- .POLYNOMIALS AND SPECIAL FUNCTIONS
- .ORDINARY DIFFERENTIAL EQUATIONS
- .INTERPOLATION, APPROXIMATION AND QUADRATURE
- .LINEAR ALGEBRA
- .PROBABILITY, STATISTICS AND TIME SERIES
- .NONLINEAR EQUATION SOLVERS

NSRDC

'NSRDC' IS A LIBRARY OF DTNSRDC WRITTEN AND/OR SUPPORTED SUBPROGRAMS. SUBROUTINES FORMERLY AVAILABLE ONLY ON TAPE CLIBRARYUPD3 ARE INCLUDED. MANY SUBPROGRAMS HAVE NEVER BEEN PART OF ANY OTHER LIBRARY.

'NSRDC' HAS A WIDE VARIETY OF ROUTINES INCLUDING, BUT NOT LIMITED TO:

- ROOTS OF POLYNOMIALS (HELP(COMPLEX), PROOT)
- SPECIAL FUNCTIONS (BESSI/J/K/Y, CELLI/ELLI, FRESNEL, GAMMA, JACOBIAN ELLIPTIC)
- INTEGRATION (KUTMER, SIMPUN, DISCOT, GAUSS QUADRATURE)
- INTERPOLATION AND CURVE FITTING (FAST FOURIER TRANSFORM, HARMONIC ANALYSIS, LEAST-SQUARES AND ORTHOGONAL POLYNOMIALS)
- EIGENSYSTEM (VARAH1/2)
- SIMULTANEOUS LINEAR EQUATIONS (CGAUSS, CMPINV, MATINS)
- CHARACTER MANIPULATION (ADJL/R, ASHIFT/SHIFTA, CENTER, COMPSTR, EXTPRM, GETCHA/PUTCHA, IFINDCH, LASTC, MOVSTR, REPLAC, CHFILL, TRAILBZ)
- SORTING (ASORT, ASORTMV, SSORT, SSORTI/F/L)
- DATE MANIPULATION (JGDATE, JULIAN, MONTH, NEWDAT, WEKDAY)
- TIME MANIPULATION (ALTIME/ELTIME, IHMS/ISEC)
- FILE MANIPULATION (CLUNLO/UNLOAD, REQUEST, ROUTE, SKPFIL, ZPFUNC (ATTACH/CATALOG/EXTEND/PURGE/ETC.))
- DEBUGGING AIDS AND DUMPING (DUMPA, PRTIME, RECOVERD)
- PLOTING (PLOTPR)
- EXTRACTING INFORMATION ABOUT JOB (CPU467, HERE, IDID, JOBNAME, JOBOG, OVLNAME)

DOCUMENTATION FOR MANY OF THE ROUTINES IN 'NSRDC' CAN BE FOUND IN CCLIB/N.

JUNE 1977

PAGE 10-6

SSP

VERSION III OF THE IBM SCIENTIFIC SUBROUTINE PACKAGE CONTAINS ONLY SINGLE PRECISION ROUTINES. ALL SUBROUTINE NAMES ARE PRECEDED BY THE LETTER I (E.G., MINV IN THE SSP MANUAL IS IMINV IN THE LIBRARY.) SINCE MANY OF THESE ROUTINES WERE OPTIMIZED FOR 32-BIT WORDS, USE WITH CAUTION ON CDC. THESE ROUTINES ARE NO LONGER MAINTAINED BY IBM.

REFERENCE: IBM FORM NUMBER GH20-0205, SYSTEM/360 SCIENTIFIC SUBROUTINE PACKAGE, VERSION III, PROGRAMMER'S MANUAL.

*** PLOTTING LIBRARIES ***

PLOTTING ROUTINES ARE AVAILABLE FOR CALCOMP PEN PLOTTERS, MICROFILM PLOTTERS OR PRINTER PLOTS.

SEVERAL LIBRARIES CONTAIN THE ROUTINES FOR THE VARIOUS CALCOMP PLOTTERS AVAILABLE.

*** PRINTER PLOTS ***

SEE CCLIB/N: PLOTPR, AND SSP MANUAL, PAGE 452 (PLOT IN THE MANUAL AND IPLOT IN THE LIBRARY).

*** OFF-LINE PLOTTERS ***

SEVERAL DIFFERENT OFF-LINE PLOTTERS ARE AVAILABLE AT DTNSRDC. SOFTWARE PACKAGES FOR EACH ARE DISCUSSED BELOW. A DEVICE-INDEPENDENT PACKAGE, WHICH CAN USE MANY DIFFERENT PLOTTERS, IS DISCUSSED ON 10-17.

*** CALCOMP PLOTTERS ***

THE BASIC CALCOMP PACKAGE FOR ON-LINE CALCOMP PLOTTERS AND OFF-LINE PLOTTERS ON THE CDC 6000 COMPUTERS INCLUDES THE FOLLOWING FORTRAN ROUTINES:

| | |
|--------|---|
| PLOT | (WITH ENTRIES: PLOTS, WHERE, FACTOR, OFFSET) CONVERT ALL PEN MOVEMENT SPECIFICATIONS FROM INCHES TO ACTUAL PLOTTER COMMANDS |
| SCALE | EXAMINE A DATA ARRAY TO DETERMINE STARTING AND SCALE VALUES AND CONVERT FROM INCHES TO ACTUAL PLOTTER COMMANDS |
| AXIS | DRAW AN AXIS LINE WITH APPROPRIATE SCALE ANNOTATIONS AND TITLE |
| LINE | PLOT A SERIES OF SCALED DATA POINTS DEFINED BY TWO ARRAYS (X AND Y) |
| NUMBER | DRAW THE DECIMAL EQUIVALENT OF AN INTERNAL FLOATING- POINT NUMBER |
| SYMBOL | DRAW ANY SEQUENCE OF ALPHAMERIC CHARACTERS |

THE CALLING SEQUENCES ARE AS DESCRIBED IN "PROGRAMMING CALCOMP PEN PLOTTERS" MANUAL.

FOR CALCOMP 936 AND 763, THE FIRST CALL MUST BE
CALL PLOTS (IBUF, NSIZE, NT)
TO DEFINE THE OUTPUT UNIT "TAPENT". THE LAST CALL MUST BE
CALL PLOT (X, Y, 999)
TO CLOSE THE FILE.

EACH OF THE LIBRARIES BELOW IS AN EDITLIB LIBRARY AND REQUIRES:
ATTACH,CALCXXX.
LDSET,LIB=CALCXXX. OR LIBRARY,CALCXXX.
WHERE XXX IS ONE OF 170, 763, 936.

*** CALCOMP 936 PEN PLOTTER ***
(CALC936)

THIS LIBRARY CONTAINS ROUTINES FROM THE BASIC PACKAGE. IN OVERLAY JOBS, SUBROUTINE PLOT MUST BE IN THE MAIN LINK: (0,0) OVERLAY. THE CALLING SEQUENCES ARE AS DESCRIBED IN "PROGRAMMING CALCOMP PEN PLOTTERS" MANUAL. AN ADDITIONAL ROUTINE NEWPEN IS AVAILABLE. ROUTINE BUFF, INCLUDED IN THIS PACKAGE, AUTOMATICALLY ALLOCATES BUFFER SPACE. THEREFORE, IT IS NOT NECESSARY TO ESTABLISH A BUFFER AREA AS IS REQUIRED FOR THE CALCOMP 763 PLOTTER. THE FILE NAME FOR THE PLOT OUTPUT FILE MAY NOT BE OVERRIDDEN ON THE LGO CARD OF THE FTN PROGRAM STATEMENT.

THESE ROUTINES MAY REQUIRE UP TO 3000 OCTAL LOCATIONS.

THE OUTPUT TAPE IS A 7-TRACK, LABELLED TAPE AT 800 OR 556 BPI.

*** 1700 CALCOMP PEN PLOTTER ***
(CALC170)

THIS LIBRARY CONTAINS THE BASIC CALCOMP PACKAGE FOR THE ON-LINE CALCOMP PLOTTER ON THE CDC 1700 AT NAVSEC AND NAVAIR. THE GRAPH OUTPUT IS WRITTEN TO FILE "PLOT", WHICH NEED NOT BE ON FTN PROGRAM STATEMENT. THIS FILE HAS AUTOMATIC DISPOSITION AT END-OF-JOB (SEE GCRM, 14-4). TO ELIMINATE THE PLOT FILE DURING CODE CHECKING, USE

LGO.

ROUTE,PLOT.

THE ROUTINE PLOTS IS A DUMMY ROUTINE FOR COMPATIBILITY WITH CALCOMP 763 AND CALCOMP 936. ADDITIONAL ROUTINES AVAILABLE ARE:

IDFRMKK PRODUCE BANNER PAGE FOR CALCOMP OUTPUT
CIRCL PRODUCE GENERAL GEOMETRIC DESIGNS, E.G., RECTANGLES,
CIRCLES, ETC.

*** CALCOMP 763 PEN PLOTTER ***
(CALC763)

THIS LIBRARY CONTAINS ROUTINES FROM THE BASIC CALCOMP PACKAGE. IN OVERLAY JOBS, SUBROUTINE PLOT MUST BE IN THE MAIN LINK: (0,0 OVERLAY). THE CALLING SEQUENCES ARE AS DESCRIBED IN "PROGRAMMING CALCOMP PEN PLOTTERS" MANUAL. THE FILE NAME FOR THE PLOT OUTPUT FILE MAY NOT BE OVERRIDDEN ON THE LGO CARD OR THE FTN PROGRAM STATEMENT.

THESE ROUTINES MAY REQUIRE UP TO 3000 OCTAL LOCATIONS.

THE OUTPUT TAPE IS A SI UNLABELLED TAPE AT 556 BPI.

*** CALCOMP FUNCTIONAL PACKAGE (PROPRIETARY) ***
(CALCFN)

THE CALCOMP FUNCTIONAL SOFTWARE LIBRARY CONSISTS OF SEVERAL CATEGORIES. THESE FORTRAN ROUTINES GENERATE CALLS TO THE BASIC SOFTWARE REQUIRING THE USER TO ALSO ATTACH AND MAKE AVAILABLE THE LIBRARY FOR THE APPROPRIATE PLOTTER (CALC170, CALC763, CALC936).

BUSINESS APPLICATIONS

AXISR - DRAWS AN AXIS WITH BUSINESS ORIENTED ANNOTATION
AXISC - DRAWS AN AXIS WITH CALENDAR MONTH ANNOTATION
BAR - DRAWS BARS FOR BAR GRAPH PLOTTING
LBAXS - DRAWS A LOGARITHMIC AXIS WITH BUSINESS ANNOTATION
LGLIN - PLOTS DATA EITHER IN LOG-LOG OR IN SEMI-LOG MODE
SCALG - PERFORMS SCALING FOR LOGARITHMIC PLOTTING
SHADE - DRAWS SHADING BETWEEN DESIGNATED LINES

DRAFTING APPLICATIONS

ARCHO - DRAWS ARROWHEADS
ARROW - DRAWS LINES TERMINATED WITH AN ARROW
CNTRL - DRAWS CENTER LINES
DIMEN - DRAWS ANNOTATED DIMENSION LINES
LABEL - DRAWS ANNOTATION BETWEEN SPECIFIED POINTS

GENERAL APPLICATIONS

CIRCLE - DRAWS CIRCLE, ARC OR SPIRAL
DASHL - DRAWS DASHED LINES CONNECTING A SERIES OF DATA POINTS
DASHP - DRAWS DASHED LINES TO A SPECIFIED POINT
ELIPS - DRAWS AN ELLIPSE OR ELLIPTICAL ARC
FIT - DRAWS A CURVE THROUGH THREE POINTS
GRID - DRAWS LINEAR GRID
POLY - DRAWS AN EQUILATERAL POLYGON
RECT - DRAWS A RECTANGLE

SCIENTIFIC APPLICATIONS

CURVX - PLOTS A FUNCTION OF X OVER A GIVEN RANGE
CURVY - PLOTS A FUNCTION OF Y OVER A GIVEN RANGE
FLINE - DRAWS A SMOOTH CURVE THROUGH A SET OF DATA POINTS
LGAXS - DRAWS A LOGARITHMIC AXIS WITH ANNOTATION
POLAR - PLOTS DATA POINTS, USING POLAR COORDINATES
SCALG - PERFORMS SCALING FOR LOGARITHMIC PLOTTING
SMOOT - DRAWS A SMOOTH CURVE THROUGH SEQUENTIAL DATA POINTS

MISCELLANEOUS

CRVPT - FITS A POLYNOMIAL CURVE TO A SET OF DATA POINTS. PLOTS
FITTED CURVE, ORIGINAL DATA WITH REFERENCE AXES AND
EQUATIONS OF CURVE

AD-A043 559 DAVID W TAYLOR NAVAL SHIP RESEARCH AND DEVELOPMENT CE--ETC F/G 9/2
COMPUTER CENTER REFERENCE MANUAL, REVISION.(U)
JUN 77 S E GOOD, D V SOMMER

DAVID W TAYLOR NAVAL SHIP RESEARCH AND DEVELOPMENT CE--ETC F/G 9/2
COMPUTER CENTER REFERENCE MANUAL, REVISION.(U)
JUN 77 S E GOOD, D V SOMMER

CMLD-77-11

NL

AD
A043 559

A043 559

END
DATE
FILMED
9-77
DDC

EXAMPLES

1. CALCOMP 936 PLOTTER

| | |
|---|-----------------------------|
| XXXX,MT1. | NAME / CODE |
| CHARGE,..... | |
| ATTACH,CALC936. | BASIC CALCOMP 936 PACKAGE |
| FTN,OPT=1. | COMPILE USER PROGRAM |
| LABEL,TAPE10,L=CALC,D=HI,W,RING,VSN=CA9999. | NOS/BE LABLO TP 556 BPI |
| LDSFT(LIB=CALC936) | SELECT USER LIBRARY |
| LGO. | LOAD + EXECUTE USER PROGRAM |
| ' 7/8/9 EOR | |
| PROGRAM PLOTPGM (...) | |
| CALL PLOTS (IRUF, 0, 10) | |
| ... | |
| CALL PLOT (0., 0., 999) | |
| END | |
| ' 7/8/9 EOR | |
| (DATA) | |
| " 6/7/8/9 EOF | |

2. CALCOMP 1700 PLOTTER

| | |
|--------------------------|-----------------------------|
| XXXX. | NAME / CODE |
| CHARGE,..... | |
| ATTACH,CALC170. | CALCOMP 1700 PLOT PACKAGE |
| FTN,OPT=1. | COMPILE USER PROGRAM |
| LDSFT(LIB=CALC170) | SELECT USER LIBRARY |
| LGO. | LOAD + EXECUTE USER PROGRAM |
| RCUTE(PLOT,DC=PT,TID=AF) | PLOT DATA ON 1700 AT NAVSEC |
| ' 7/8/9 EOR | |
| (SOURCE PROGRAM) | |
| ' 7/8/9 EOR | |
| (DATA) | |
| " 6/7/8/9 EOF | |

3. CALCOMP 763 PLOTTER (ALSO USE ROUTINES IN FUNCTIONAL PACKAGE)

| | |
|------------------------------------|-----------------------------|
| XXXX,MT1. | NAME / CODE |
| CHARGE,..... | |
| FTN,OPT=1. | COMPILE USER PROGRAM |
| ATTACH,CALC763. | BASIC CALCOMP 763 PACKAGE |
| ATTACH,CALCFN. | CALCOMP FUNCTIONAL PACKAGE |
| REQUEST,TAPE16,HI,PIRG,VSN=CA9999. | NOS/BE UNLABELLED TAPE 556 |
| LDSFT(LIB=CALC763/CALCFN) | SELECT USER LIBRARIES |
| LGO. | LOAD + EXECUTE USER PROGRAM |
| ' 7/8/9 EOR | |
| PROGRAM PLOTPGM (...) | |
| DIMENSION IRUF(1000) | |
| CALL PLOTS (IRUF, 1000, 16) | |
| ... | |
| CALL PLOT (0., 0., 999) | |
| END | |
| ' 7/8/9 EOR | |
| (DATA) | |
| " 6/7/8/9 EOF | |

*** SC4020 TO CALCOMP ***
(SCCALC)

SCCALC IS A PLOT LIBRARY WHICH ALLOWS A PROGRAM WRITTEN WITH SC4020 CALLS TO PLOT ON THE CALCOMP 936 OR 763. INSTEAD OF LIBRARY SC4020, THREE LIBRARIES ARE USED: SCCALC, CALCFN AND CALC936 (OR CALC763). SEE THE EXAMPLE BELOW.

SEVERAL SC4020 ROUTINES NOT NORMALLY REFERENCED BY USERS ARE DUMMY ROUTINES. NO ID FRAME WILL BE DRAWN. THE OPTION FOR USER-WRITTEN XMODV AND YMODV FOR NON-LINEAR CONVERSION IS NOT AVAILABLE; HOWEVER, LOG PLOTS MAY BE DRAWN.

SCCALC INCLUDES ENTRY POINT NAMES BPLTV, CHARGC, DLC1 AND COMMON BLOCK DLCPLOT. THESE NAMES, AS WELL AS THE BASIC ROUTINE NAMES LISTED ON CCRM, 10-9, MAY NOT BE USER ROUTINE NAMES.

THE OUTPUT IS ON MAGNETIC TAPE TAPE48, WHICH MAY NOT BE OVERRIDDEN ON THE LGO.

EXAMPLE

| | NAME / CODE |
|---|-----------------------|
| XXXX,MT1,.... | |
| CHARGE,.... | |
| FTN,OPT=1. | |
| LABEL,TAPE48,L=XXXXCALC,D=HI,W,RING,VSN=CA9999. | |
| ATTACH,CALCFN. | |
| ATTACH,CALC936. | ** OR ATTACH,CALC763. |
| ATTACH,SCCALC. | |
| LDSET,LIB=SCCALC/CALC936/CALCFN. | ** .../CALC763/... |
| LGO. | |
| ' 7/8/9 EOR | |
| PROGRAM MYPROG (INPUT=128, OUTPUT=128, TAPE48, ...) | |
| (REST OF FTN SOURCE PROGRAM WITH SC4020 CALLS) | |
| ' 7/8/9 EOR | |
| (DATA, IF ANY) | |
| " 6/7/8/9 EOF | |

*** CALCOMP THREE-D SOFTWARE *** (PROPRIETARY)
(CALC3D)

THE THREE-D PROGRAM AUTOMATICALLY DRAWS THREE DIMENSIONAL REPRESENTATIONS OF DATA THAT CAN BE EXPRESSED AS A FUNCTION OF TWO VARIABLES.

SURFACES CAN BE DRAWN TRANSPARENT, WITH ALL LINES SHOWN, OR AS OPAQUE, WITH HIDDEN LINES REMOVED. THE SURFACE FUNCTION CAN BE SPECIFIED AS AN EXTERNAL FUNCTION TO BE EVALUATED OVER A GIVEN X AND Y RANGE, OR AS Z VALUES AT THE MESH POINTS OF A RECTANGULAR ARRAY.

THREE-D CAN AUTOMATICALLY GRID IRREGULARLY SPACED INPUT DATA. THE USER MAY INSERT ADDITIONAL DATA INTO THE ORIGINAL DATA ARRAY TO PROVIDE A SMOOTHER PLOT OF ROUGH ARRAY DATA. THE PACKAGE PROVIDES FOR AUTOMATIC ORIENTATION OF ANNOTATION TO FIT VIEWING DISTANCE AND ANGLE. THE DRIVER PROGRAM IS CONTROLLED BY USER SPECIFIED PARAMETERS ON INPUT CARDS. ALSO STEREOSCOPIC PAIRS CAN BE PRODUCED IN ACCORDANCE WITH USERS MANUAL.

THE THREE-D PROGRAM MAY BE EXECUTED BY USING THE CATALOGUED PROCEDURE CALC3D. (SEE CCLIB/P)

AT DTNSRDC, IF FIELD INLU ON THE B CARD IS SPECIFIED, IT MUST BE 3 (DATA ON TAPE3).

REFERENCE: THREE-D - A PERSPECTIVE DRAWING SOFTWARE SYSTEM, CALCOMP, 1969.

*** SC-4020 MICROFILM PLOTTER ROUTINES ***
(SC4020)

THE STROMBERG-CARLSON 4020 MICROFILM PLOTTER ROUTINES (SCORS) ARE IN LIBRARY SC4020. THE FORTRAN PROGRAM CARD MUST CONTAIN THE FILE TAPE48. THE OUTPUT IS EITHER THE USER'S OWN STRANGER TAPE AT 556 BPI OR ON THE SPECIAL FILMPL FILE FOR BATCH OUTPUT. THE SCORS REFERENCE MANUAL IS AVAILABLE IN CODE 1892.

THE FIRST PLOTTING CALL SHOULD BE TO CAMRAV AND IDFRMV TO PLOT ID FRAME. THE LAST CALL SHOULD BE CALL PLTND (0).

*** SC-4020 - USER INFORMATION ***

1. MOST SUBROUTINE CALLS FROM SCORS PACKAGE FOR SC 4020 MAY BE USED DIRECTLY ON THE CDC 6000. THE PROGRAMMERS REFERENCE MANUAL FOR THE SC 4020 COMPUTER RECORDER BY STROMBERG-CARLSON CORPORATION IS STILL VALID DOCUMENTATION, EXCEPT FOR THE FOLLOWING CURRENTLY KNOWN CORRECTIONS AND ADDITIONS.
 - A. THE USER MAY SUPPLY A TAPE FOR EACH JOB. THIS TAPE WILL BE PROCESSED ON THE CMLD SC 4060 WHEN THE REQUEST FORM IS SUBMITTED TO ADP CONTROL AFTER THE CDC 6000 RUN.
 - B. BATCHED OUTPUT FOR THE 4020 PACKAGE MAY BE WRITTEN TO FILE FILMPL BY INCLUDING FILMPL,TAPE48=FILMPL ON THE PROGRAM CARD OR OVERRIDING THE PROPER FILE POSITION ON THE LGO. STANDARD REQUESTS FOR PROCESSING THIS OUTPUT MUST BE SUBMITTED TO ADP CONTROL CENTER WITH THE SEVEN-CHARACTER JOB NAME INSTEAD OF A TAPE NUMBER.
2. TO PROPERLY GENERATE AN ID FRAME BOTH
CALL CAMRAV (N)
CALL IDFRMV (NAME, CODE, PHONE)
MUST APPEAR IN EITHER ORDER AS THE FIRST CHARACTERON ROUTINES CALLED.
NAME - UP TO 10 ALPHAMERIC CHARACTERS
CODE - UP TO 10 ALPHAMERIC CHARACTERS
PHONE - UP TO 10 ALPHAMERIC CHARACTERS
THESE ARGUMENTS ARE NORMALLY DEFINED AS HOLLERITH DATA.
3. CALL PLTND(0) WILL EMPTY THE OUTPUT BUFFER, AND PLACE AN END-OF-FILE MARK ON THE SC 4020 OUTPUT FILE. IT SHOULD BE THE LAST CHARACTERON ROUTINE CALLED IN EVERY JOB. THE FRAME COUNT IS ENTERED IN THE DAYFILE AND ON OUTPUT FILE. (CALL PLTND(1) DOES NOT ISSUE FRAME COUNT OR WRITE END FILE.)
4. THE FOLLOWING ROUTINES ARE NOT AVAILABLE:
SCOUTV
VCPS
TABL4V, TABL6V
TABL5V, TAB05V
TABL7V, TAB07V
TABL8V, TABL9V
TAB11V, TAB15V
FRMNOV, TPNUMV

5. SINCE MOST CDC 6000 SC 4020 ROUTINES ARE FORTRAN CODED THERE IS ONLY ONE LENGTH FOR EACH CALLING SEQUENCE:

- A) FRAMEV MUST HAVE 1 ARGUMENT. USE CALL FRAMEV(0).
- B) XAXISV AND YAXISV MUST HAVE 3 ARGUMENTS. IF THE AXIS IS TO GO TO END OF FRAME, USE 0 OR 1023 FOR 3RD ARGUMENT.
- C) POINTV HAS ONLY THE 3 ARGUMENT FORM USING FLOATING-POINT DATA VALUES. THE EQUIVALENT OF THE 4 ARGUMENT RASTER FORM IS CALL ROINTV(IX1,IY1,NC).
- D) NOFRV HAS ONLY 1 ARGUMENT TO RETURN THE TOTAL FRAME COUNT.
- E) KWKPLT MAY BE CALLED ONLY WITH THE 3 ARGUMENT FORM. NO TITLES ARE DISPLAYED.
- F) VCHARV MUST HAVE 6 ARGUMENTS. THE ALTERNATE SINGLE PATTERN SPECIAL OCTAL DATA FORM OF VCHARV WITH 7 ARGUMENTS IS NOT AVAILABLE.
- G) PRINTV MUST BE CALLED WITH 4 ARGUMENTS -
CALL PRINTV (N,BCDXTX,IX,IY)
- H) RESETV, BRITEV, FAINTV, RESCLV HAVE NO ARGUMENTS.

6. IF OVERLAY IS USED, FLAGSV AND PLOTDD (WHICH ARE INTERNALLY USED BY THE CALLS IN THE USER PROGRAM) MUST NOT BE OVERLAID. TO FORCE LOADING OF THESE INTO THE 0,0 OVERLAY, USE THE FOLLOWING CONTROL CARDS:
LDSET,LIB=SC4020,USE=FLAGSV/PLOTDD.
LGO.

7. CALL INTBCD (I,N) IS A NEW ROUTINE WHICH CONVERTS AN INTEGER I (LESS THAN 10000) INTO ITS BCD EQUIVALENT IN N, RIGHT JUSTIFIED AND SPACE FILLED.

8. ALL ALPHAMERIC TITLES MUST BE SINGLE PRECISION ARRAYS (10 CHARACTERS PER WORD IN DISPLAY CODE) WHICH WILL BE INTERNALLY CONVERTED TO SC 4020 BCD. ALL VECTORS OF COEFFICIENTS FOR PLOTTING MUST BE SINGLE PRECISION REAL NUMBERS.

9. THE PLOT TAPE WILL BE 7-TRACK STRANGER TAPE, 556 BPI, ON TAPE48. IF A LABELLED TAPE IS USED, F=S IS REQUIRED. A USER MAY WRITE IN OTHER THAN FILE 1 OF HIS TAPE IF DESIRED, BUT MULTIPLE FILES CANNOT BE WRITTEN ON THE SAME TAPE BY INDEPENDENT JOBS SUBMITTED NEAR THE SAME TIME, SINCE NO JOB SEQUENCING IS GUARANTEED.

LABEL,TAPE48,L=XXXXPLOT,F=S,D=HI,W,VSN=CA9999,RING.
REQUEST,TAPE48,HI,S,VSN=CA9999,RING.

IF THE FILMPL FILE IS USED FOR 4020 OUTPUT, NO TAPE IS REQUIRED.

10. ROUTINES FNPLT, LGCHAR, CURVE AND SIXFN, WHICH CALL ON THE SCORS PACKAGE, HAVE BEEN ADDED TO SC4020.

*** SAMPLE 4020 USAGE ***

USER'S OWN TAPE

```

USERSC,CM50000,T300,MT1,P3.                NAME / CODE
CHARGE,USER,JJJJJJJJJJ.
FTN.                ** MOUNT TAPE ONLY AFTER GOOD COMPILE
LABEL,TAPE48,L=USER4020,D=HI,F=S,W,VSN=CA9999,RING.    OR    REQUEST
ATTACH,SC4020.
LDSET,LIB=SC4020.
LGC.
'  7/8/9      EOR
    PROGRAM PLGRD(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE48,...)
(CWN FORTRAN PROGRAM)
'  7/8/9      EOR
(DATA CARDS)
"  6/7/8/9    EOF

```

BATCH PLOT QUEUE

```

USERB,CM50000,P3,T300.                NAME / CODE
CHARGE,USER,JJJJJJJJJJ.
FTN,CPT=1.
ATTACH,SC4020.
LDSET,LIB=SC4020.
LGC(,,FILMPL)                ** OVERRIDE THIRD FILE NAME
'  7/8/9      EOR
    PROGRAM PLTES(INPUT,OUTPUT,TAPE48,...)
(CWN FORTRAN PROGRAM)
'  7/8/9      EOR
(DATA CARDS)
"  6/7/8/9    EOF

```


*** SC4060 - META USER INFORMATION ***
(SC4060)

META LANGUAGE FOR SC4060 IS OUTPUT BY INTEGRATED GRAPHICS SOFTWARE, A PACKAGE OF 26 SUBROUTINES FOR GRAPHING, BASIC GEOMETRY, STRINGS OF TEXT, AND NUMERIC DATA. THE MODE ARRAY, A 200 WORD COMMUNICATION REGION IN THE USER PROGRAM SHOULD NOT BE OVERLAYED.

TO PROPERLY GENERATE AN ID FRAME
CALL IDFRMG (Z, NAME, CODE, PHONE)
WHERE Z IS THE MODE ARRAY AND THE OTHER ARGUMENTS ARE HOLLERITH DATA UP TO 8 CHARACTERS.

EACH FRAME ADVANCE IS PROGRAMMED BY CALL PAGEG(Z,0,1,1)

TO EMPTY THE BUFFERS AT END OF RUN CALL EXITG(Z)

BATCHED OUTPUT FOR THE META PACKAGE MAY BE WRITTEN TO FILE FILMPR BY INCLUDING FILMPR,TAPE10=FILMPR ON THE PROGRAM CARD OR OVERRIDING THE PROPER FILE POSITION ON LGO.

USERMT,CM50000,P3,T100.

NAME / CODE

CHARGE,USER,JJJJJJJJJJ.

FTN.

ATTACH,SC4060.

** META LIBRARY

LDSET,LID=SC4060.

LGO(,,,FILMPR)

** OVERRIDE FOURTH FILE NAME

* 7/8/9 FOR

PROGRAM USEIT(INPUT,OUTPUT,TAPE7,TAPE10,TAPE5=INPUT)

(USER PROGRAM)

* 7/8/9 EOR

(USER DATA)

" 6/7/8/9 EOF

*** DISSPLA (PROPRIETARY) ***

THE DISPLAY INTEGRATED SOFTWARE SYSTEM AND PLOTTING LANGUAGE (DISSPLA) IS A PACKAGE OF MACHINE INDEPENDENT SUBROUTINES WHICH MAY INTERFACE WITH THE DTNSRDC CALCOMP PLOTTERS, SC4060, IGS274, AND TEKTRONIX TERMINALS OR MAY CREATE A DEVICE INDEPENDENT FILE FOR LATER POST-PROCESSING ON ANY PLOTTING DEVICE.

1. DIRECT USE OF DISSPLA IN BATCH JOBS REQUIRES THREE EDITLIB LIBRARIES:

ATTACH,DISSPLA.

ATTACH,NSRDC.

ATTACH,PLOTLIB. WHERE PLOTLIB IS CALC936,
CALC763, CALC170, SC4060, TEK48, TEK30.

WHEN LOADING THE USER PROGRAM, BEGIN THE LOAD SEQUENCE WITH:

LDSET,LIB=DISSPLA/NSRDC/PLOTLIB.

CORE REQUIREMENTS WILL BE FROM 70000 TO 120000.

2. FOR THE SC4060 INTERFACE, THE FIRST DISSPLA REFERENCE IN USER PROGRAM IS:

CALL SC4060 (8HNAME ,8HCODE ,8HTEL)

PLOT OUTPUT MAY BE WRITTEN TO:

USERS OWN TAPE (TAPE10 ON FORTRAN PROGRAM CARD)

FILMPR (FILMPR,TAPE10=FILMPR ON PROGRAM CARD)

SC4060 PROCESSING MUST BE BY THE "META" PROGRAM. (SEE CCRM 10-22)

3. FOR THE CALCOMP INTERFACES, THE FIRST DISSPLA REFERENCE IN USER PROGRAM IS:

CALL CALCMP (10)

USER MUST SUPPLY HIS OWN TAPE FOR TAPE10. (SEE CCRM 10-14)

ONLY THE CALC763 CALCOMP INTERFACE ROUTINE NEEDS TAPE10 ON THE FORTRAN PROGRAM CARD.

CALC170 CREATES FILE "PLOT" WHICH MUST BE RUN FROM OR ROUTED TO THE 1700 BATCH TERMINALS AF OR AG. "PLOT" NEED NOT BE ON THE FORTRAN PROGRAM CARD.

4. THE IGS274 INTERFACE REQUIRES ONLY THE FIRST TWO ATTACH CARDS OF 1. ABOVE. WHEN LOADING THE 274 USER PROGRAM, BEGIN THE LOAD SEQUENCE WITH:

LDSET,LIB=DISSPLA/NSRDC/IGS274.

THE FIRST DISSPLA REFERENCE IN USER PROGRAM IS:

CALL IGS274 (NCON) WHERE NCON IS THE CONSOLE
NUMBER (1, 21B, OR 31B).

NO INTERACTION IS POSSIBLE WITH THE DRAWINGS CONSTRUCTED ON 274 SCOPE EXCEPT BY DATA IN THE BATCH JOB.

5. THE DIRECT TEKTRONIX INTERFACES CANNOT GENERALLY BE USED DUE TO EXCESSIVE CORE REQUIREMENTS. THE FIRST DISSPLA REFERENCE IN USER PROGRAM IS:

CALL TEKTRN (IBAUD) WHERE IBAUD IS 480 OR
300.

UNDER INTERCOM, THE TEKTRONIX IS NORMALLY USED INDIRECTLY (SEE 6. AND 8. BELOW).

6. WHEN IT MAY BE NECESSARY TO OUTPUT A GIVEN SET OF PLOTS ON MORE THAN ONE TYPE OF PLOTTING DEVICE, TO ALLOW FOR INTERACTIVE VIEWING ON THE TEKTRONIX TERMINALS, OR TO ALLOW FOR LATER EDITING OF THE PLOT INFORMATION: A DEVICE INDEPENDENT INTERFACE CAN PREPARE COMPRESSED OUTPUT ON FILE "PLFILE" FOR USE BY ANY POST-PROCESSORS. ONLY THE FIRST TWO ATTACH CARDS OF 1. ABOVE ARE NEEDED. THE USER PROVIDES FOR "PLFILE" BY:

REQUEST,PLFILE,*PF.

TO LOAD THE USER PROGRAM, BEGIN THE LOAD SEQUENCE WITH:

LDSET,LIB=DISSPLA/NSRDC.

"PLFILE" NEED NOT BE ON THE FORTRAN PROGRAM CARD.

THE FIRST DISSPLA REFERENCE IN USER PROGRAM IS:

CALL COMPRS

CATALOG "PLFILE" FOR LATER PROCESSING WITH ONE OF THE POST-PROCESSORS.

7. AFTER THE DEVICE INTERFACE ROUTINE CALL, THE NEXT DISSPLA REFERENCE MUST BE CALL BGNPL (N) WHERE N IS THE PLOT NUMBER. EACH PLOT ENDS WITH CALL ENDPL (N) TO PROPERLY FLUSH THE PLOT OUTPUT BUFFERS, THE LAST DISSPLA REFERENCE IN ALL USER PROGRAMS MUST BE:

CALL DONEPL

8. POST-PROCESSORS CONTAINED ON EDITLIB FILE "DISPOST" INCLUDE POP4060, POP274, TEK480, TEK300.

TEKTRONIX POST-PROCESSORS MAY BE USED INTERACTIVELY.

TO ACCESS A POST-PROCESSOR:

ATTACH,PLFILE,.... COMPRESSED PLOT DATA

ATTACH,DISPOST.

LIBRARY,DISPOST.

PRECEDUE USE OF "TEK480." BY TYPING:

SCREEN(80,66)

PRECEDUE USE OF "TEK300." BY TYPING:

FILE,TAPE62,RT=S.

OPTIONAL DATA CARDS ALLOW FOR WINDOWING, SCALING, SELECTIVE PLOTTING, CHANGING POSITION, OR SUPERIMPOSING PLOTS.

THE FIRST DATA CARD FOR "POP4060." MUST CONTAIN NAME, CODE AND TELEPHONE, EACH LEFT-ADJUSTED IN AN A10 FIELD (3A10).

REFERENCES:

DISSPLA POCKET MANUAL

DISSPLA BEGINNER/INTERMEDIATE MANUAL VOLUME I

DISSPLA ADVANCED MANUAL VOLUME II

DISSPOP (DISSPLA POST-PROCESSOR OPTION)

*** LIBRARIES OF MAIN PROGRAMS ***

MANY FREQUENTLY USED PROGRAMS AND MAJOR PACKAGES OF PROGRAMS HAVE BEEN PUT INTO LIBRARIES. EACH WILL REQUIRE ATTACH AND LDSET/LIBRARY CARDS TO ACCESS THE DESIRED PROGRAM.

TO ACCESS ANY OF THESE PUBLIC LIBRARIES:

| | |
|--------------------|--|
| ATTACH,LIBNAME. | WHERE LIBNAME IS A LIBRARY DESCRIBED BELOW |
| LDSET(LIB=LIBNAME) | OR LIBRARY(LIBNAME) |
| PROG. | NAME OF PROGRAM ON LIBNAME TO BE EXECUTED |

THE CURRENTLY AVAILABLE LIBRARIES ARE:

BIMED - SELECTED ROUTINES FROM UCLA BIO-MEDICAL STATISTICAL PROGRAMS
UTILITY - A WIDE VARIETY OF LOCAL UTILITY PROGRAMS

BIMED

THIS LIBRARY CONTAINS TESTED ROUTINES OF THE UCLA BIO-MEDICAL STATISTICAL PROGRAMS. OTHERS WILL BE ADDED AS REQUESTED. BIMED IS AVAILABLE ON THE 6700/6600 AND CURRENTLY CONTAINS:

| | |
|---|--------|
| | --CM-- |
| BMD03D - CORRELATION WITH ITEM DELETION | 105000 |
| BMD05D - GENERAL PLOT INCLUDING HISTOGRAM | 60000 |
| BMD05M - DISCRIMINANT ANALYSIS FOR SEVERAL GROUPS | 70000 |
| BMD03R - MULTIPLE REGRESSION WITH CASE COMBINATIONS | 70000 |
| BMD04R - PERIODIC REGRESSION AND HARMONIC ANALYSIS | 70000 |
| BMD05R - POLYNOMIAL REGRESSION | 45000 |
| BMD07R - NON-LINEAR LEAST SQUARES | 62000 |
| BMD12V - MULTIVARIATE ANALYSIS OF VARIANCE AND COVARIANCE | 110000 |

BIMEDP

THE BIMEDP BIO-MEDICAL STATISTICAL PROGRAMS FROM UCLA ACCEPT DATA WITH PARAMETER LANGUAGE CONTROL SIMILAR TO SPSS.

THESE PROGRAMS ARE CURRENTLY AVAILABLE ON TAPE.

UTILITY

'UTILITY' IS A LIBRARY OF DTNSRDC WRITTEN AND/OR MAINTAINED PROGRAMS. THEY MAY BE EXECUTED BY ONE OF THE FOLLOWING:

- A) ATTACH,UTILITY.
 LDSET,LIB=UTILITY. OR LIBRARY,UTILITY.
 PROG,..... ** PROGRAM TO BE EXECUTED
- B) BEGIN,UTILITY,,PROG,.....

ROUTINES 'CCRM' AND 'PAGEPRT', FOR EXTRACTING PORTIONS OF THIS MANUAL, ARE LISTED BELOW. DOCUMENTATION FOR ALL ROUTINES IN 'UTILITY' MAY BE FOUND IN CCLIB/U.

CCRM EXTRACT ALL PAGES FROM COMPUTER CENTER REFERENCE MANUAL WHICH WERE MODIFIED AFTER USER-SPECIFIED DATE

TO USE - CCRM,OUTPUT,MONTH,YEAR,PM.

OUTPUT - LISTABLE OUTPUT

MONTH - FIRST 3-4 CHARACTERS OF MONTH (E.G., JAN, FEB, SEPT, ETC.) OR THE CORRESPONDING DIGITS (1, ..., 12)

YEAR - 2 OR 4 DIGITS OF YEAR (E.G., 77, 1977)
 (ALL PAGES MODIFIED AFTER MONTH, YEAR WILL BE PRINTED)

PM - OMITTED OR - DO NOT PUT 'PM' 'NOPM' MESSAGES IN OUTPUT
 ANYTHING ELSE - PUT THE MESSAGE 'PM 1-PART NARROW UNLINED PAPER' AT START OF OUTPUT AND 'PM RESTORE PRINTER' AT THE END
 (USE ONLY FOR PRINTING AT TERMINALS WHICH SUPPORT 'PM' MESSAGES.)

LOAD FL - 300008

DEFAULT - CCRM,OUTPUT,0,0,NOPM.
 (WILL PRINT ENTIRE MANUAL BUT IS NOT EFFICIENT)

EXAMPLE - TO PRINT ALL REVISIONS:
 CCRM,OUTPUT,MAY,77.

REMARKS - TITLE AND 'R' (REVISION SUMMARY) PAGES ARE ALWAYS PRINTED.

PAGEPRT

PRINT SELECTED PAGE(S) FROM A DOCUMENT

TO USE - PAGEPRT,INFILE,ID,OUTPUT,REW,PAGE,INPUT.

- INFILE - INPUT DOCUMENT FILE, ONE OF:
 - .1-7 CHAR PFN (PROG WILL ATTACH)
 - .1-7 CHAR LFN (FILE IS ATTACHED)
 - .OMITTED (COMPUTER CENTER REF MAN)
- ID - ONE OF
 - .1-7 CHAR ID (USE IF INFILE IS PFN)
 - .0 (IF FILE IS ALREADY ATTACHED)
 - .OMITTED (SYSTEM PUBLIC ACCESS ID)
- OUTPUT - LISTABLE OUTPUT
- REW - INFILE REWIND OPTION
 - A - REWIND AFTER USE
 - B - REWIND BEFORE USE
 - R - REWIND BEFORE AND AFTER USE (DEFAULT)
 - U - REWIND BEFORE, REWIND AND UNLOAD AFTER USE
 - OTHER - NO REWIND
- PAGE - SINGLE PAGE DESIRED (1-7 CHAR)
 - OR-
 - IF OMITTED, READ PAGE NUMBERS FROM FILE INPUT.
- INPUT - DATA CARD INPUT FILE (ONLY IF PAGE PARAMETER OMITTED)

MAX FL - 300008

DEFAULT - PAGEPRT,*,**,OUTPUT,REWIND,,INPUT.

- * - COMPUTER CENTER REFERENCE MANUAL
- ** - SYSTEM PUBLIC ACCESS ID

REMARKS - IF SINGLE PAGE DESIRED HAS 8, 9 OR 10 CHARACTERS, IT MUST BE ENTERED ON A DATA CARD.

DATA - USED ONLY IF PAGE PARAMETER OMITTED.
ONE OR MORE CARDS WITH DESIRED PAGE NUMBERS
(ONE PER CARD) IN COLUMNS 1-10 OF EACH CARD.

EXAMPLE - PRINT PAGE 4-2 OF COMPUTER CENTER REF MANUAL:
PAGEPRT,,,\$4-2\$.

PRINT PAGES 1-9 AND 1-10 OF CCRM.

PAGEPRT.

" 7/8/9 EOR

1-9

1-10

" 6/7/8/9 EOF

*** MISCELLANEOUS PROGRAMS ***

THE FOLLOWING PROGRAMS ARE NOT IN ANY LIBRARIES BUT MUST BE INDIVIDUALLY ATTACHED (SEE CCLIB):

CVT360 A SNOBOL PROGRAM TO CONVERT DOUBLE PRECISION S/360 FORTRAN PROGRAMS TO SINGLE PRECISION CDC FORTRAN.

COPYBFR RECREATE A CDC "RANDOM FILE" FROM DATA COPIED EARLIER TO A SEQUENTIAL FILE, OR CREATE A COPY OF A RANDOM FILE. IT MAY BE USED TO RECREATE A PROPER OLDPL IF COPYBF WAS USED ERRONEOUSLY. TO USE
 ATTACH,COPYBFR.
 COPYBFR,INFILE,OUTFILE.

COPYS A GENERAL PURPOSE UTILITY FROM NORTHWESTERN UNIVERSITY WHICH PROVIDES A LARGE VARIETY OF COPY OPERATIONS FOR SEQUENTIAL OR RANDOM FILES.

*** VIM PROGRAMMING PACKAGES ***

LARGE SYSTEMS ORDERED FROM VIM ARE OFTEN ON SEPARATE TAPE FILES.

REFERENCE

| | | |
|---------|--|----------------------------|
| ARRIBA | ALL-INTEGER PROGRAMMING SYSTEM FOR SMALL INTEGER PROGRAMMING PROBLEMS INCLUDING SEVERAL SOLUTION ALGORITHMS. | H1 UTEX ARIBA |
| CIVCC | CIVIL ENGINEERING COMPUTATION SYSTEM: A PACKAGED PROGRAM DESIGNED TO CALCULATE COMPLEX AND COMPOUND CURVES AND SURFACES IN LARGE GEOMETRY PROBLEMS. | |
| ECAP | ELECTRONIC CIRCUIT ANALYSIS PROGRAM: INTEGRATED SYSTEM OF PROGRAMS TO ASSIST IN DESIGN AND ANALYSIS OF ELECTRICAL CIRCUITS. A GRAPHICAL VERSION WAS OBTAINED FROM NWL. | T4 CODA ECAP |
| NOVACOM | A COMPUTER PROGRAM FOR NON-ORTHOGONAL ANALYSIS OF VARIANCE AND COVARIANCE. | NWL TR-2108 NWL TR-2137 |
| STRESS | STRUCTURAL ANALYSIS LANGUAGE. SEE USERS AND REFERENCE MANUALS FOR DATA PREPARATION AND PROGRAM DESIGN. | |

*** OTHER SOURCES OF PROGRAMS ***

A COLLECTION OF BASIC ROUTINES WAS OBTAINED FROM NOL.

SHARE FORTRAN LIBRARY ROUTINES (ON FILE FROM IBM 7090).

SOME MEDIUM SPEED REMOTE TERMINALS USE A HOUSTON PLOTTER PACKAGE.

SPECIAL PURPOSE COMPLETE PROGRAMS AND PROGRAMMING LANGUAGES WHICH WERE PURCHASED FROM OTHER ORGANIZATIONS ARE DISCUSSED IN CHAPTER 11.

USERS MAY CONTRIBUTE ROUTINES TO THE LIBRARY.

ROUTINES MAY BE ORDERED FROM VIM, THE CDC 6000 USERS GROUP. REQUESTS FOR VIM ROUTINES SHOULD BE MADE TO CODE 1892.

*** PROGRAM DOCUMENTATION ***

DOCUMENTATION FOR PROGRAMS, SUBPROGRAMS AND PROCEDURES (CCRM, 7-21) CAN BE OBTAINED FROM CODE 1892. SOME DOCUMENTATION IS AVAILABLE FROM THE SYSTEM AND IS OBTAINED BY USING PROGRAM PROGDOC IN LIBRARY UTILITY (SEE CCLIB/P: PROGDOC) WITH THE FOLLOWING GENERAL CONTROL CARD:

BEGIN,UTILITY,,PROGDOC,LIB,,PROG.

WHERE LIB IS ONE OF

UTILITY - PROGRAMS ON LIBRARY UTILITY

NSRDC - SOME ROUTINES ON LIBRARY NSRDC

OTHER - SOME MISCELLANEOUS ROUTINES

PROFIL - PUBLIC-ACCESS PROCEDURES (CCRM, 7-21; CCLIB/P)

PROG IS ONE OF

NAME - LIST THIS DOCUMENT

ALL - LIST ALL DOCUMENTS

MACHINE-READABLE DOCUMENTS DESCRIBING PROGRAMS, SUBPROGRAMS AND PROCEDURES ARE PRINTED IN CCLIB, CCLIB/N, CCLIB/P AND CCLIB/U.

*** SOURCE CODE ***

THE SOURCE CODE FOR MANY ROUTINES IS AVAILABLE IN UPDATE LIBRARIES FOR USER MODIFICATION. SEE EXAMPLE ON CCRM, 7-6 (TOP). NOTE THAT VSN AND LABEL CARDS WILL BE DIFFERENT FOR EACH TAPE (SEE BELOW).

| LIBRARY | MACHINE | SOURCE CODE LOCATION |
|---------|-----------|---|
| BIMED | 6700/6600 | LABEL,LIBR,L=CLIBBIMED,D=HY,R,VSN=CA0278,NORING. |
| | 6400 | LABEL,LIBR,L=CLIBBIMED,D=HY,R,VSN=CJ0347,NORING. |
| NSRDC | 6700/6600 | ATTACH,OLDPL,NSRDCPL,ID=CSYS. |
| | | LABEL,LIBR,L=CLIBRARYUPD3,D=HY,R,VSN=CA0279,NORING. |
| | 6400 | LABEL,LIBR,L=CLIBRARYUPD3,D=HY,R,VSN=CJ0348,NORING. |
| SC4020 | 6700/6600 | LABEL,LIBR,L=CLIBRARYUPD3,D=HY,R,VSN=CA0279,NORING. |
| | 6400 | LABEL,LIBR,L=CLIBRARYUPD3,D=HY,R,VSN=CJ0348,NORING. |
| SSP | 6700/6600 | LABEL,LIBR,L=CLIBRARYUPD3,D=HY,R,VSN=CA0279,NORING. |
| | 6400 | LABEL,LIBR,L=CLIBRARYUPD3,D=HY,R,VSN=CJ0348,NORING. |
| UTILITY | 6700/6600 | ATTACH,OLDPL,UTILITYPL,ID=CSYS. |

***** SPECIAL PURPOSE PROGRAMMING LANGUAGES *****

THIS CHAPTER DISCUSSES VARIOUS PROGRAMMING LANGUAGES AND PACKAGES AVAILABLE ON THE DTNSRDC 6000 COMPUTERS.

*** ALGOL ***

'ALGOL420' IS A PUBLIC-ACCESS PERMANENT FILE. EXECUTION OF AN ALGOL-60 PROGRAM IS BY THE FOLLOWING SEQUENCE OF CONTROL CARDS (REFERENCES ON THIS PAGE ARE TO THE CDC ALGOL VERSION 4 REFERENCE MANUAL, PUBLICATION NUMBER 60384700 OR 60496600).

| | |
|-------------------------------------|--|
| JOB CARD WITH AT LEAST CM46000 | NAME / CODE |
| CHARGE,..... | |
| ATTACH,ALGOLIB,ALGOL420. | USE THIS LFN (ALGOLIB) |
| LIBRARY,ALGOLIB. | |
| ALGOL. | SEE CHAPTER 6 FOR OPTIONAL PARAMETERS |
| RFL,NNNNNN. | NNNNNN IS THE JOB CARD CM (USUALLY REQUIRED) |
| LGO. | |
| • 7/8/9 FOR | |
| (ALGOL MAIN PROGRAM AND PROCEDURES) | |
| • 7/8/9 FOR | |
| (SET OF CHANNEL CARDS) | |
| (DATA CARDS) | |
| • 7/8/9 FOR | |
| • 6/7/8/9 EOF | |

THE MAIN PROGRAM AND EACH PROCEDURE EXCEPT THE LAST MUST BE FOLLOWED BY AN 'EOP' CARD (4-4). THE SET OF CHANNEL CARDS IS REQUIRED ONLY WHEN CHANNELS OTHER THE STANDARD (60 FOR INPUT, 61 FOR OUTPUT) ARE USED. IF THERE ARE NEITHER CHANNEL CARDS NOR DATA CARDS, THE 6/7/8/9 CARD MAY FOLLOW THE LAST END CARD.

NOTE - ALL ALGOL KEYPUNCHING SHOULD BE DONE ON 026 KEYPUNCHES. THE SQUARE BRACKET SYMBOLS ARE BEST IMPLEMENTED BY "(/" AND "/)".

THE FOLLOWING HINTS MAY PROVE USEFUL. THE EASIEST WAY TO OUTPUT VARIABLES AND ARRAYS IS VIA "OUTREAL" AND "OUTARRAY" PROCEDURES (3-25). THESE MAY ALSO BE USED TO OUTPUT INTEGER VARIABLES AND ARRAYS USING THE REAL FORMAT.

THE CDC IMPLEMENTATION OF ALGOL DIFFERS FROM THE PUBLICATION LANGUAGE IN REGARD TO THE FORMAL PARAMETERS OF A PROCEDURE. ALL PARAMETERS OF A PROCEDURE MUST BE DECLARED APPROPRIATELY. FOR EXAMPLE, IN MANY PUBLISHED ALGORITHMS FORMAL PARAMETERS WHICH ARE THE NAMES OF PROCEDURES ARE NOT DECLARED (2-49: 5.4.5). ALSO, ALL PROCEDURES CALLED IN A PROGRAM MUST BE DECLARED VIA THE "CODE" SYMBOL (2-50). ALGOL AND FORTRAN PROGRAMS MAY CALL EACH OTHER (2-50).

EACH ALGOL SYMBOLIC WORD IN THE REFERENCE LANGUAGE IS REPRESENTED IN CDC BY "WORD", WHERE " (QUOTE) IS THE 4-8 PUNCH.

*** AUTOMATICALLY PROGRAMMED TOOLS (APT) ***

THE APT SYSTEM IS USED TO PREPARE PUNCHED PAPER TAPES FOR NUMERICALLY CONTROLLED MACHINE TOOLS. FROM INPUT LANGUAGE DESCRIBING A PART TO BE MACHINED, APT GENERATES THE DETAILED COMMANDS NECESSARY TO DIRECT THE MOTIONS OF A NC MACHINE TOOL. USING THIS SYSTEM, EVEN COMPLEX PARTS CAN BE MACHINED QUICKLY, ACCURATELY, AND INEXPENSIVELY.

IN APT3, AN EXTENSION TO APT CALLED FMILL IS AVAILABLE FOR MACHINING OF ARBITRARILY SHAPED THREE-DIMENSIONAL SURFACES SUCH AS FOUND ON MODEL PROPELLERS AND MODEL SHIP HULLS. IT ACCEPTS AS INPUT ANY RECTILINEAR MESH OF X-Y-Z COORDINATES. POST PROCESSORS IMPLEMENTED WITH APT3 INCLUDE: GECENT, BENDIX, COLTMC.

A NEWER VERSION OF APT (APT4) IS ALSO AVAILABLE ON PERMANENT FILE. THERE ARE NO POST PROCESSORS INSTALLED IN APT4. POST PROCESSING IS POSSIBLE USING AN OPTION UNDER APT3.

APT3 SAMPLE SETUP

| | |
|-------------------------|-----------------------------|
| JOBNAME,CM65200. | NAME / CODE |
| CHARGE,XXXX,JJJJJJJJJJ. | |
| COPYCR,INPUT,INFILE. | |
| REWIND,INFILE. | |
| ATTACH,APT3. | ** APT3 SYSTEM |
| APT3,INFILE. | ** EXECUTE APT3 |
| ATTACH,PUNTAP. | ** PAPER TAPE PUNCH PROGRAM |
| PUNTAP. | ** PUNCH PAPER TAPE |
| * 7/8/9 EOR | |
| (APT SOURCE CARDS) | |
| * 7/8/9 EOR | |
| (PUNTAP DATA CARD) | |
| ** 6/7/8/9 EOF | |

APT4 SAMPLE SETUP

| | |
|--|-----------------|
| JOBNAME,CM122000. | NAME / CODE |
| CHARGE,XXXX,JJJJJJJJJJ. | |
| ATTACH,APT4. | ** APT4 SYSTEM |
| APT4. | ** EXECUTE APT4 |
| (PUNTAP MAY BE EXECUTED HERE AS ABOVE) | |
| * 7/8/9 EOR | |
| (APT SOURCE CARDS) | |
| ** 6/7/8/9 EOF | |

REFERENCE: AUTOMATICALLY PROGRAMMED TOOLING SYSTEM (APT IV) VERSION 2
REFERENCE MANUAL, CDC PUB. NO. 17326900.

*** CHECKPOINT/RESTART ***

CHECKPOINT DUMPS MAY BE TAKEN BY THE PROGRAMMER PERIODICALLY DURING LONG RUNNING JOBS (FOR EXAMPLE, OVER 30 MINUTES WALL CLOCK TIME). THEN RESTART IS USED TO REINITIATE THE JOB FROM A DUMP IN EVENT OF PROGRAM OR SYSTEM FAILURE. CHECKPOINT DUMPS MAY BE CALLED BY A CONTROL CARD OR FROM A USER PROGRAM. (SEE NOSBE, 4-12, 4-72; SCOPE, CHAPTER 10)

| | | |
|--------------|--------------------|--------------|
| CONTROL CARD | CKP. | |
| FORTRAN EXT. | CALL CHEKPTX(IVAR) | WHERE IVAR=0 |

UNLESS THE USER REQUESTS OR LABELS A TAPE WITH CHECKPOINT DISPOSITION, THE SYSTEM WILL REQUEST A SCRATCH TAPE FOR THE DUMPS. THE JOB CARD MUST ALLOW FOR THIS MT. EACH USER SHOULD PROVIDE FOR HIS OWN CHECKPOINT TAPE. THE CK PARAMETER IS REQUIRED ON ANY LABEL OR REQUEST CARD FOR A CHECKPOINT REEL.

LABEL,CCCCCCC,L=XXXXCKPT,X=CK,D=HY,RING,VSN=CB9999,W.
(USE ABOVE LABEL CARD FOR INITIAL CHECKPOINT RUN.
FOR RESTART, CHANGE PARAMETER 'W' TO 'R'.)

THE SYSTEM USES FILE NAMES CCCCCC, CCCCCI, CCCCCM, AND CCCCCO.

IF ANY DEVICE SET IS USED IN THE JOB, USER MUST 'MOUNT' THE DEVICE SET BEFORE THE 'RESTART' CARD. IF ANY TAPES ARE USED IN THE JOB, USER SHOULD SUPPLY 'VSN' CARD(S) (NOT LABEL OR REQUEST) BEFORE 'RESTART'.

| | |
|---------------------|---|
| RESTART,FILENM,NUM. | |
| WHERE | FILENM IS CHECKPOINT TAPE (DEFAULT: CCCCCC) |
| | NUM CHECKPOINT DUMP TO RESTART FROM (DEFAULT: 1) |

IF THE RESTART TAPE IS A MULTI-REEL FILE, IT IS POSSIBLE TO RESTART FROM OTHER THAN THE FIRST TAPE.

CHECKPOINTS MAY NOT BE TAKEN IN A PROGRAM USING RANDOM ACCESS (READMS) FILES UNLESS THEY ARE PREVIOUSLY CATALOGED.

IF A MULTI-STEP JOB TAKES A CHECKPOINT BETWEEN STEPS OF THE JOB, THE RESTART WILL COMMENCE EXECUTING CONTROL CARDS AT THE IMMEDIATELY NEXT STEP AFTER THE 'CKP.' CARD. NO ALTERATION OF 'INPUT' CONTROL CARDS OR DATA RECORDS IS ACCEPTED, AS NO CARDS IN THE RESTART JOB STREAM WILL BE LOOKED AT AFTER THE 'RESTART.' CARD. FOR A MULTISTEP JOB TO CHECKPOINT ONLY IN CASE OF ERROR (AFTER EXIT), A BEGIN/REVERT PROCEDURE MUST FOLLOW THE 'CKP.'.

CHECKPOINT EXAMPLES - FTR

```

PROGRAM SGLONG (INPUT,OUTPUT,TAPE6=OUTPUT,TAPE5=INPUT)
DATA IVAR/0/
B=TIME(DUM)
DECODE(10,5,B) IH,IM,IS
ITIME = IH*3600 +IM*60 +IS
C  ENTER LONG PROGRAM LOOP
    DO 500 K=1,1000
        . . .
        . . .
        . . .
C  TAKE CHECKPOINT DUMP EVERY 20 CLOCK MINUTES
5  FORMAT(3(1X,I2))
    B1=TIME(DUM)
    DECODE(10,5,B1) IH,IM,IS
    INOW = IH*3600 +IM*60 +IS
    IF(INOW-ITIME .LT. 1200) GO TO 15
    CALL CHEKPTX(IVAR)
C  RECALCULATE STARTING TIME IN CASE OF A RESTART
    B=TIME(DUM)
    DECODE(10,5,B) IH,IM,IS
    ITIME = IH*3600 +IM*60 +IS
15  CONTINUE
500  CONTINUE
    STOP
    END

```

```

PROGRAM SGLONG (INPUT,OUTPUT,TAPE6=OUTPUT,TAPE5=INPUT)
DATA IVAR/0/
B=SECOND(DUM)
C  ENTER LONG PROGRAM LOOP
    DO 500 K = 1,1000
        . . .
        . . .
        . . .
C  TAKE CHECKPOINT DUMP EVERY 360 CPU SECONDS
    B1=SECOND(DUM)
    IF (B1-B .LT. 360.) GO TO 15
    CALL CHEKPTX(IVAR)
C  SET NEXT TIME INTERVAL
    B=B1
15  CONTINUE
500  CONTINUE
    STOP
    END

```

NOTE-- BY DELETING *** CARDS ABOVE, COULD CHECKPOINT EACH LOOP

*** OPERATOR CHECKPOINT ***

AT END OF SHIFT OR WHEN A LONG RUNNING JOB MUST BE TERMINATED,
THE CURRENT STATUS OF THE JOB MAY BE PRESERVED ON TAPE FOR RESTART AT A
LATER TIME BY OPERATOR ACTION.

1. N.LOCKIN. TO LOCKIN THE JOB
2. N.CHECKPT. TO SAVE THE CURRENT STATUS OF JOB ON TAPE
3. N.ASSIGN XX LOAN THE USER AN UNLABELLED SCRATCH TAPE.
PASTE ON A LABEL WITH JOBNAME AND DATE.
HOLD 24 HOURS. (E.G. CALMKDR, 6/30/77)
4. N.COMMENT. RESTART JOB USING TAPE . SEE USER SERVICES
N.COMMENT. DROPPING JOB DUE TO -----.
5. N.DROP. TO VACATE CONTROL POINT.
6. USER MAY RESTART THE JOB BY PUTTING IN A 4 CARD JOB.

```

XXXX,CMNNNNNN,MT1,TNNN.      NAME / CODE
CHARGE,XXXX,JJJJJJJJJJ.
VSN,CCCCCCC=XXXXYY.
RESTART,CCCCCCC,1.
" 6/7/8/9 EOF

```

WHERE TNNN IS TIME REQUIRED TO COMPLETE JOB
CMNNNNNN IS CORE REQUIRED TO RUN THE JOB
MT MUST ALLOW FOR MOUNTING THE RESTART TAPE
XXXXYY IS THE FIRST 6 CHARACTERS OF THE JOB NAME BEING
RESTARTED, AS PASTED ON SCRATCH TAPE.

FINAL OUTPUT OF THE RESTARTED JOB WILL INCLUDE ALL OUTPUT
COMPUTED AND PRINTED BEFORE THE RESTART WAS BEGUN.

*** COMRADE - COMPUTER-AIDED DESIGN ENVIRONMENT ***

THE COMRADE EXECUTIVE, FUNCTIONING UNDER INTERCOM, ALLOWS SYSTEM DESIGNERS TO CONSTRUCT APPLICATION SUBSYSTEMS WHICH CAN BE USED THROUGH NON-COMPUTER ORIENTED PROCEDURES. EACH SUBSYSTEM CONSISTS OF INTERRELATED "COMMAND PROCEDURES" WHICH AUTOMATICALLY INVOKE A SEQUENCE OF COMPUTER ACTIONS. THESE PROVIDE FOR TUTORIAL MESSAGES, DATA INPUT REQUESTS, DYNAMIC PROGRAM INITIATION, CONTROL CARD EXECUTION, AND SUBMISSION OF BATCH JOBS.

COMRADE PROVIDES CAPABILITIES FOR CONTROLLING ACCESS TO COMMANDS AND FILES, AND FOR REPORTING ON SYSTEM USAGE. ALSO INCLUDED IS A DATA MANAGEMENT SYSTEM WHICH ALLOWS THE DEFINITION, UPDATING AND QUERYING OF COMPLEX DATA STRUCTURES USING EITHER FORTRAN-CALLABLE SUBROUTINES OR USER-ORIENTED COMMAND PROCEDURES.

THE FOLLOWING SUBSYSTEMS ARE CURRENTLY WITHIN COMRADE:

- ISDS
- PERSONNEL
- SHARP
- SYST
- GENERAL

FOR FURTHER INFORMATION, CONTACT BILL GORHAM, CODE 1828, (202) 227-1285.

*** COMPASS ***

THE COMPASS ASSEMBLER IS USED TO MAINTAIN MANY OF THE NOS/BE SYSTEM ROUTINES FOR CPU AND PPU. SOURCE STATEMENTS INCLUDE PSEUDO-INSTRUCTIONS, MACROS AND MICROS AS WELL AS SYMBOLIC MACHINE INSTRUCTIONS.

*** DATA MANAGEMENT SYSTEM 170 (DMS170) ***

DMS170 IS CONTROL DATA'S IMPLEMENTATION OF A COMPREHENSIVE DATA MANAGEMENT SYSTEM. AMONG THE MAJOR COMPONENTS OF DMS170 ARE:

DATA DESCRIPTION LANGUAGE (DDL)

THIS COMPILER DESCRIBES THE DATA BY CREATING SCHEMAS (WHICH DESCRIBE THE ORGANIZATION AND FULL RECORD CONTENT OF THE DATA BASE) AND SUB-SCHEMAS (WHICH DESCRIBE SETS OF DATA ELEMENTS TO BE OPERATED ON VIA COBOL OR Q/U). IT IS CDC'S IMPLEMENTATION OF THE CODASYL DATA BASE TASK GROUP DATA DESCRIPTION LANGUAGE.

QUERY UPDATE (Q/U)

THIS SPECIAL ENGLISH-LIKE LANGUAGE ALLOWS RETRIEVAL, MANIPULATION AND UPDATE OF DATA BASE ELEMENTS AND INCLUDES REPORT WRITER CAPABILITY. IT CAN ACCESS DATA USING DDL SUB-SCHEMA OR DIRECTLY USING Q/U DESCRIPTION OF FILE. IT ALLOWS BATCH OR INTERACTIVE SESSIONS. (NOTE: CURRENT CORE REQUIREMENTS OF APPROXIMATELY 70000 WORDS RESTRICTS USE TO BATCH ONLY.)

CYBER DATABASE CONTROL SYSTEM (CDCS)

THIS HANDLES DATA MAPPING, DATA VALIDATION, TRANSACTION LOGGING, AND DATA BASE PROCEDURE LINKING.

DATA BASE UTILITIES

UTILITIES ARE PROVIDED FOR LOGGING AND DATA BASE RECOVERY.

DMS170 PROVIDES HIGH LEVEL DATA BASE MANAGEMENT FEATURES SUCH AS DATA SECURITY, TRANSPARENCY OF DATA ORGANIZATION TO END USER, LOGGING AND RECOVERY, ETC. WHILE USING STANDARD CDC RECORD MANAGER TECHNIQUES FOR ACTUAL DATA FILE ORGANIZATION.

REFERENCES FOR FURTHER INFORMATION.

| | |
|---|------------------|
| DMS170 DATA ADMINISTRATORS USER'S GUIDE | 60499100 |
| DDL REFERENCE MANUAL VOL. 1, 2, 3 | 60498400,500,600 |
| QUERY UPDATE 3 REFERENCE MANUAL | 60498300 |
| QUERY UPDATE 3 USER'S GUIDE | 60387700 |
| CDCS REFERENCE MANUAL | 60498700 |
| DATA BASE UTILITIES | 60498800 |

*** MARS VI ***

MARS VI VERSION 2.1 IS A RETRIEVAL SYSTEM DESIGNED FOR FULL INTERFACE WITH HIGH LEVEL PROGRAMS (FORTRAN, COBOL). MARS ORDERS DATA BASE FILES IN TWO WAYS.

- (1) PARTIAL INVERSION
- (2) FULL INVERSION

BOTH DATA FILE ORGANIZATIONS CAN BE USED UNDER VERSION 2.1 OF MARS.

PARTIAL INVERSION ALLOWS FOR COMPLETE HIGH LEVEL PROGRAM INTERFACE. IT WORKS ON FILES WHICH CAN BE CREATED BY EITHER FORTRAN OR COBOL. THE "RECORD" FILE PRODUCED BY MARS CAN BE READ BY FORTRAN/COBOL. THERE ARE THREE MAJOR FILES TO BE CONSIDERED IN USING MARS-

- (1) DATA FILE - INITIAL INPUT FILE
GENERATED IN COBOL OR FORTRAN
- (2) RECORD FILE - SEQUENTIAL RETRIEVAL FILE, SIMILAR TO
ORIGINAL DATA FILE WITH DIFFERENT BLOCKING.
- (3) LOAD/SAVE FILE - KEY FILE CONTAINING RANDOMLY ACCESSIBLE
KEYS SELECTED FROM THE DATA FILE AT
LOADER TIME.

UNDER FULL INVERSION ALL DATA IS 'INVERTED', OR MADE RANDOMLY ACCESSIBLE. THE FULLY INVERTED MARS VI FILE IS NOT USABLE WITH FORTRAN OR COBOL, SINCE IT IS A FILE OF KEYS AND BEARS NO RESEMBLANCE TO THE ORIGINAL DATA FILE. IN ADDITION, SINCE ALL ITEMS ARE INVERTED, THIS FILE FORM REQUIRES MANY MORE PRU'S FOR STORAGE. THIS FULLY INVERTED FORMAT SHOULD BE AVOIDED IF POSSIBLE. IT HAS, HOWEVER ONE ADVANTAGE: IT CAN OPERATE ON VARIABLE LENGTH INPUT RECORDS AND ON A VARIABLE NUMBER OF LOWER LEVELS IN A HIERARCHIC STRUCTURE.

MARS VI EXECUTION

SINCE MARS IS A PUBLIC-ACCESS PERMANENT FILE, THE FOLLOWING CONTROL CARD IS REQUIRED:

ATTACH,MARS.

THE FIRST USER DIRECTIVE TO MARS MUST BE:
USERLIB IS MARS:

MARS VI CREATION CONCEPTS

MARS VI (BOTH FULL AND PARTIAL INVERSION) HAS THREE MAJOR MODULES.

- (1) DEFINE
- (2) LOADER (PARTIAL INVERSION) INVERTER (FULL INVERSION)
- (3) RETRIEVAL

MARS REQUIRES THAT AN INITIAL DATA FILE BE DEFINED, PASSED BY EITHER THE LOADER OR INVERTER IN ORDER TO CREATE A RETRIEVAL FILE. IN THE CASE OF PARTIAL INVERSION A RECORD FILE IS ALSO GENERATED WHICH CAN BE ACCESSED SEQUENTIALLY. ALL FIELDS PRESENT ON THE INITIAL DATA FILE WILL BE ON THE RECORD FILE, INCLUDING KEY ITEMS. KEY ITEMS WILL ALSO BE ON THE LOAD/SAVE FILE (RANDOM FILE). HENCE, IT IS IMPORTANT THAT ITEMS BE NAMED KEYS ONLY IF THEY WILL BE ACCESSED FREQUENTLY.

THERE IS NO NEED TO CATALOG OR USE MARS (PF) SINCE LOAD DATA BASE ON LFN CREATES A COPY OF YOUR RANDOM KEY FILE. IN ADDITION ONLY ONE PERMANENT FILE IS USED INSTEAD OF EIGHT. THE USE OF THE LOAD/SAVE FILE WILL SAVE SYSTEM PRU'S SINCE EIGHT PF'S REQUIRE 8*56 PRU'S TO BE ALLOCATED BY THE SYSTEM, EVEN IF ONLY 2 OR 3 ARE ACTUALLY USED IN EACH.

MARS VI RETRIEVAL CONCEPTS

UNDER MARS VI, ALWAYS USE THE SAME INFORMATION WHEN REFERENCING THE RECORD FILE, I. E., 'RECORD FILE LFN;'. THE INFORMATION TO BE USED IS IS THE NAME GIVEN THE FILE AT LOAD TIME.

MARS IV CAN GENERATE SEQUENTIAL FILES WITH PART OF THE DATA FROM THE ORIGINAL FILE VIA THE REPORT FILE IS LFN; AND THE SUBSET VERB. THE FILE GENERATED BY MARS CAN BE READ BY COBOL WITH RECORDING MODE DECIMAL OR BY FORTRAN WITH FORMATTED READ STATEMENTS. IF MORE THAN 150 CHARACTERS PER RECORD, AN ADDITIONAL PARAMETER MUST BE USED ON THE FTN PROGRAM CARD. IF DATA IS BUFFERED IN, ARRAY SHOULD BE 512 WORDS AND DEBLOCKING MUST BE DONE BY THE USER.

*** MARS VI SAMPLE JOB STREAMS ***

CREATE A PARTIAL INVERSION FILE

```

USERMAR,CM60000,T1000,P3.
CHARGE,USER,JJJJJJJJJJ.
ATTACH,INFILE,ID=USER.
REQUEST,SDATA,*PF.
REQUEST,RECFILE,*PF.
ATTACH,MARS.
MARS.
CATALOG,SDATA,ID=USER,XR=RDONL.
CATALOG,RECFILE,ID=USER,XR=RDONL.
" 7/8/9 EOR
USERLIB IS MARS;
DEFINE:
NEW DATA BASE IS TEST:
1) PERSON(NAME):
2) CODE(NAME):
MAP:
DEFINE SUBITEM;
RECORD ID IS TESTA;
COL 1-20,C1,PERSON,KEY;
COL 21-26,C2,CODE,KEY;
COL 27-37,ADDRESS;
COL 38-40,DATA.NUM;
END DEFINITION;
LOADER:
DATA FILE IS INFILE; IF COBOL-GENERATED USE "INFILE,CB;"
RECORD FILE IS RECFILE;
SCAN TESTA;
SAVE DATA BASE ON SDATA;
EXIT:
" 6/7/8/9 EOF

```

RETRIEVE FROM EXISTING FILE

```

USERMAR,CM60000,T500.
CHARGE,USER,JJJJJJJJJJ.
ATTACH,RECFILE,ID=USER.
ATTACH,SDATA,ID=USER.
REQUEST,TEMP,*PF.
ATTACH,MARS.
MARS.
CATALOG,TEMP,OUTPUTFILE,ID=USER,XR=KEEP.
" 7/8/9 EOR
USERLIB IS MARS;
LOAD DATA BASE FROM SDATA;
RETRIEVAL:
RECORD FILE IS RECFILE;
IF DATA EQ 10 PRINT PERSON;
REPORT FILE IS TEMP;
IF DATA LT 5 SUBSET PERSON, CODE;
REPORT FILE IS OUTPUT;
REPORT PERSON WHERE DATA EXIST;
EXIT:
" 6/7/8/9 EOF

```


*** NASTRAN ***

NASTRAN IS A GENERAL PURPOSE FINITE ELEMENT STRUCTURAL ANALYSIS PROGRAM, CAPABLE OF PERFORMING A WIDE RANGE OF ANALYSIS ON MODELS OF COMPLEX STRUCTURES. SOME OF THE CAPABILITIES OF NASTRAN INCLUDE STATIC STRESS ANALYSIS, NATURAL FREQUENCY ANALYSIS, BUCKLING ANALYSIS, FREQUENCY RESPONSE ANALYSIS AND TRANSIENT RESPONSE ANALYSIS.

THE VERSION OF NASTRAN CURRENTLY STORED ON SYSTEM PERMANENT FILES IS LEVEL 16.0. THE PREVIOUS LEVEL (15.2 - NAVY-NASTRAN) IS ONLY AVAILABLE ON THE 6400 AS CYCLE 1.

EXECUTION OF THE PROGRAM MAY BE INVOKED BY THE FOLLOWING SEQUENCE OF CONTROL CARDS:

| | |
|------------------------|---|
| JOB CARD | WITH CM, MT, T SET AS NEEDED FOR SIZE OF DATA |
| CHARGE,.... | |
| VSN,.... | AS NEEDED FOR TAPES |
| LABEL,NPTP,0=HI,.... | OPTIONAL-FOR RUNS WITH CHKPNT YES |
| LABEL,OPTP,0=HI,R,.... | OPTIONAL-FOR RESTARTS |
| REQUEST,PLT2,HI,S. | OPTIONAL-FOR PLOTTING |
| MAP,OFF. | |
| ATTACH,NASTRAN. | |
| RFL,NNNNNN. | NNNNNN IS JOB CARD CM |
| NASTRAN.ATTACH | |
| " 7/8/9 EOR | |
| (DATA CARDS) | |
| " 6/7/8/9 EOF | |

THE PROGRAM REQUIRES A MINIMUM OF 140000 OCTAL OF CENTRAL MEMORY. HOWEVER, MEMORY REQUESTS SHOULD BE ADJUSTED TO PROBLEM SIZE FOR EFFICIENT SOLUTION. ADDITIONAL INFORMATION ON THE NASTRAN PROGRAM AND RELATED PROGRAMS IS AVAILABLE FROM THE NUMERICAL STRUCTURAL MECHANICS BRANCH, CODE 1844, TELE. 227-1938 OR 227-1660.

BANDIT ALSO REQUIRES RFL.

*** MIMIC ***

MIMIC IS A DIGITAL/ANALOG SIMULATION LANGUAGE FOR SOLVING SYSTEMS OF ORDINARY DIFFERENTIAL EQUATIONS. MIMIC, A CONTINUOUS SYSTEM SIMULATOR, IS A PARALLEL LANGUAGE SINCE THE STATEMENTS DEFINING A PROGRAM MAY BE WRITTEN IN ANY ORDER. THE ORIGINAL LANGUAGE WAS DEVELOPED BY THE AIR FORCE. TO USE:

ATTACH,MIMIC.
MIMIC.

*** PERT ***

PROGRAM EVALUATION AND REVIEW TECHNIQUE (PERT) MAY BE USED TO PLAN, MONITOR AND EVALUATE PROJECTS AND PROGRAMS. PERT/TIME PROVIDES PROJECT CONTROL FROM THE STANDPOINT OF TIME. TO USE:

ATTACH,PERT66,PERTTIME.
RFL,120000.
PERT66.

PERT/COST PROVIDES EFFECTIVE PROJECT COST CONTROL. TO USE:

ATTACH,PERTC.
RFL,112000.
PERTC.

*** QWICK QWERY (PROPRIETARY) ***

THE QWICK QWERY SYSTEM CAN PROVIDE THE USER WITH AN ANALYSIS OF ANY FILE (BINARY OR CODED) ON THE CDC 6000. A FILE MAY BE STORED ON TAPE, DEVICE SET PACK, OR ON A PERMANENT FILE. THE USER SETS UP A DICTIONARY WHICH RELATES AN ATTRIBUTE NAME USED BY QWICK QWERY TO SPECIFIC WORDS OR CHARACTERS ON THE FILE. THEN THE USER CAN QUERY THE FILE AND OBTAIN INFORMATION ABOUT SPECIFIC ATTRIBUTES. ONE EXAMPLE WOULD BE TO PROVIDE THE NAMES AND JOB DESCRIPTIONS OF ALL PERSONNEL EMPLOYED OVER THE PAST TEN YEARS. THIS INFORMATION COULD BE LISTED SEPARATELY FOR EACH DIVISION, WITH THE PERSON WITH THE LONGEST TENURE LISTED FIRST. THE USER COULD SEARCH AN OUTPUT FILE FROM AN ANALYSIS PROGRAM TO PRINT THE RESULTS FROM RUNS THAT MEET CERTAIN CRITERIA.

REFERENCE MANUAL - A QUICK LOOK AT QWICK QWERY
ASSISTANCE WITH INTERFACING FILES - LORRAINE MINOR, CODE 1892.3
CLARENCE PERRY, CODE 1822

*** RIQS (PROPRIETARY) ***

REMOTE INFORMATION QUERY SYSTEM (RIQS), WRITTEN AT NORTHWESTERN UNIVERSITY, INCLUDES FACILITIES FOR FILE CREATION, LIMITED RETRIEVAL, SEARCH AND UPDATE OF A DATA BASE. THE RIQS SYSTEM CAN CREATE A DATA BASE OF LIKE-STRUCTURED RECORDS OF MEDIUM SIZE. NUMERICAL CALCULATIONS CAN BE PERFORMED ON RECORDS. PRINTED REPORTS AND INDEXES CAN BE GENERATED. RIQS OFFERS SOME CAPABILITIES AVAILABLE IN MARS VI.

RIQSONLINE IS A COMPANION SYSTEM FOR USE FROM AN INTERACTIVE TERMINAL AND ALLOWS FILE SEARCHING, SPSS FILE CREATION, AND SIMPLE PRINTER PLOT GENERATION.

RIQS WAS OBTAINED BY ONE OF OUR CUSTOMERS (THE CONSTRUCTION ENGINEERING RESEARCH LABORATORY (CERL), COMPUTER SCIENCES BRANCH). QUESTIONS AND COMMENTS SHOULD BE DIRECTED TO USER SERVICES.

*** SHARP DATA BASE MANAGEMENT SYSTEM ***

SHARP, SHIPS ANALYSIS AND RETRIEVAL PROGRAM, IS A GENERALIZED DATA BASE MANAGEMENT SYSTEM (DMS) DEVELOPED AT THE NAVAL SHIP RESEARCH AND DEVELOPMENT CENTER (DTNSRDC), CARDEROCK, MARYLAND. SHARP IS ON THE CDC 6600 COMPUTER UNDER NOS/BE 1.0 AND INTERCOM, WHICH PROVIDES TIME-SHARING ACCESS TO THE SYSTEM. THE COMRADE MONITOR IS USED TO CONTROL THE EXECUTION OF SHARP.

SHARP IS A SELF-CONTAINED DMS, DESIGNED TO ALLOW NON-TECHNICAL PERSONS TO DEFINE, BUILD, MAINTAIN, AND INTERROGATE DATA BASES WITHOUT REQUIRING APPLICATION PROGRAM INTERFACES. SHARP ALLOWS ON-LINE AND OFF-LINE ACCESS TO DATA BASES. USER-ORIENTED, ENGLISH-LIKE LANGUAGE IS USED FOR BOTH RETRIEVAL AND REPORT GENERATION AND IS HIGHLY SUITED FOR INTERACTIVE USE FROM REMOTE TERMINALS. APPLICATION PROGRAM INTERFACES ARE AVAILABLE IF NEEDED FOR USER PROGRAMS TO ACCESS DATA IN THE DATA BASE AND FOR GENERATING SPECIAL REPORTS ON DATA RETRIEVED DURING A SHARP INTERROGATION OF THE DATA BASE.

CURRENT DATA BASE APPLICATIONS UNDER THE SHARP DMS INCLUDE:

- . TECHNICAL DOCUMENT (BIBLIOGRAPHIC)
- . SHIP OVERHAUL EXPERIENCE
- . SHIP OPERATING EXPERIENCE
- . COMPUTER CENTER EXPENSE ACCOUNTING
- . LIBRARY CIRCULATION
- . SYSTEM SOFTWARE ACCOUNTING
- . OCEAN SCIENCE RESEARCH PROJECT INFORMATION

THE MAJOR FEATURES OF SHARP INCLUDE:

- . GENERALIZED DATA BASE DEFINITION
- . GENERALIZED USER-ORIENTED FORMAT CHECK CAPABILITY
- . PARTIALLY INVERTED FILE STRUCTURE
- . VARIABLE LENGTH AND VARIABLE DATA RECORD STRUCTURE
- . ABILITY TO CHANGE DATA BASE DEFINITION WITHOUT REBUILDING DATA BASE
- . ON-LINE FILE MAINTENANCE WITH FREE FORM DIRECT ENTRY CAPABILITY
- . ENGLISH-LIKE, USER-ORIENTED LANGUAGE FOR QUERY COMMANDS REPORT DEFINITIONS, AND FORMAT CHECKING
- . INTERACTIVE REPORT DEFINITION WITH ABILITY TO CATALOG DEFINED REPORTS FOR FUTURE SELECTION
- . ON-LINE INTERACTIVE AND BATCH RETRIEVAL
- . ON-LINE AD HOC COMPUTATION CAPABILITY
- . HOST LANGUAGE INTERFACE
- . MULTI-LEVEL SORT CAPABILITY

SHARP WAS DEVELOPED IN COBOL TO FACILITATE MODIFICATION AND CONVERSION TO OTHER COMPUTERS. IN ADDITION TO THE CDC 6000, SHARP IS OPERATIONAL ON THE UNIVAC SERIES 70/45 COMPUTER SYSTEMS AT DPSCFAC, SAN DIEGO AND DPSCFANT, NORFOLK. THE SYSTEM IS BEING CONVERTED TO IBM SYSTEM/370/155 AND BURROUGHS 4700 COMPUTER SYSTEMS.

REFERENCE MANUAL SHARP DATA BASE MANAGEMENT SYSTEM
USER INFORMATION MANUAL TM-188-76-1
FOR ASSISTANCE CONTACT BEN WALLIS, CODE 1822, (202) 227-1533

*** SORT/MERGE 4.0 ***

SORT/MERGE IS A FLEXIBLE SYSTEM ROUTINE FOR SEQUENCING AND COMBINING FILES. IN ADDITION TO THE USUAL ALPHANUMERIC FIELDS, SORT MAY BE ON FLOATING-POINT, FIXED-POINT (INTEGER) AND LOGICAL TYPE FIELDS. A USER OPTION ALLOWS CHANGE OF SORT SEQUENCE.

THE SORT/MERGE SYSTEM MAY BE CALLED DIRECTLY AS A STAND-ALONE SYSTEM WITH CONTROL CARDS TO ESTABLISH FILES AND SORT FIELDS. IN COBOL, THE SORT VERB USES THE SYSTEM AS A SUBROUTINE. FORTRAN HAS A SET OF SUBROUTINES WHICH MAY BE USED TO CALL SORT/MERGE WITHIN AN FTN PROGRAM.

GUIDELINES FOR USE OF CONTROL CARD VERSION OF SORT/MERGE:

1. THE SORTMRG CONTROL CARD HAS OPTIONS FOR REWINDING THE DIRECTIVE AND LIST FILES. REWIND ALL OTHER FILES CONCERNED IN THE SORT (UNLESS SPECIAL POSITIONING OF AN EXISTING FILE IS NEEDED) BEFORE SORTING.
2. SORT/MERGE USES RECORD MANAGER AS ITS INPUT/OUTPUT PROCESSOR. FILE CARDS ARE REQUIRED FOR ALL FILES TO BE SORTED OR MERGED AND FOR THE RESULTING OUTPUT FILE.
3. INPUT IS A VALID FILE FOR SORT INPUT. THE SPECIAL SYSTEM FILES INPUT, OUTPUT AND PUNCH DO NOT REQUIRE FILE CONTROL CARDS. THEY ALWAYS HAVE DEFAULT BT=C AND RT=Z.
4. OUTPUT IS A VALID FILE FOR SORT OUTPUT. THE FIRST CHARACTER OF EACH RECORD IS USED FOR CARRIAGE CONTROL.
5. IF NO INPUT FILE CARD IS SUPPLIED, THE USER MUST SPECIFY AN EXIT 1 ON THE "OWNCODE" DIRECTIVE AND PROVIDE A ROUTINE TO READ THE INPUT RECORDS.
6. IF SORT/MERGE DIRECTIVE CARDS ARE NOT IN THE INPUT STREAM, BUT ARE ALREADY STORED AS UNIT RECORDS ON FILE "ABC", USE SORTMRG(I=ABC) OR SORTMRG(I=ABC/R) TO INDICATE THE SOURCE OF THE SORT DIRECTIVES. (THE SECOND FORM WILL REWIND "ABC" BEFORE READING THE DIRECTIVES.) IF THE DIRECTIVES ARE ON THE UPDATE "COMPILE" FILE, USE SORTMRG(I) OR SORTMRG(I/R) .
7. JOB CARD CM SHOULD BE AT LEAST 12000 OCTAL MORE THAN IS NEEDED TO RUN THE COBOL OR FTN PROGRAM WHICH USES SORT/MERGE INTERNALLY.
8. SORT/MERGE 3.0 DIRECTIVES ARE ACCEPTED IF THE PARAMETER "6C" IS ADDED TO THE SORTMRG CONTROL CARD (I.E., SORTMRG(6C)). FILE CARDS ARE NOT REQUIRED WITH SORT/MERGE 3.0 CONTROL CARDS. THE VERSION 3.0 "RECORD" DIRECTIVES ARE CONVERTED INTERNALLY INTO CORRESPONDING RECORD MANAGER FILE CARDS.
9. RECORDS WITH IDENTICAL SORT KEYS ARE SEQUENCED ARBITRARILY UNLESS THE "RETAIN" PARAMETER IS ADDED TO THE "OPTIONS" DIRECTIVE. WHEN "RETAIN" IS SPECIFIED, IDENTICAL KEYS ARE SEQUENCED IN THE ORDER IN WHICH THEY WERE READ.

*** SAMPLE SORT/MERGE SETUPS ***

1. SORT CARD IMAGE TAPE ON ENDPUNCHING (COLUMNS 73-80)

```

USERST,CM60000,MT1.                                NAME / CODE
CHARGE,USER,JJJJJJJJJJ.
LABEL,CARDS,L=USERCARDS,D=HY,R,VSN=CA7777,NORING.
REQUEST,SCRATCH,*PF.                                ** SCRATCH FILE ON DISK
FILE(CARDS,MRL=80,BT=C,RT=Z)
FILE(SCRATCH,MRL=80,BT=C,RT=Z)
SORTMRG.
COPYSF(SCRATCH,OUTPUT)                                ** LIST SORTED FILE
* 7/8/9      EOR
SORT
FILE,INPUT=CARDS(CU),OUTPUT=SCRATCH
FIELD,ENDPCH(73,8,DISPLAY)
KEY,ENDPCH(A,COROL6)
END
" 6/7/8/9    EOF

```

2. CREATE AND SORT UNIT RECORDS. RECORDS ARE SORTED ON 23-25 (ASCENDING) AND 2-16 (DESCENDING). NOTE: SORTING COULD HAVE BEEN DONE WITHIN THE FTN PROGRAM.

```

JOBNAME,.....                                NAME / CODE
CHARGE,.....
FTN.
LGO.
FILE(TAPE7,MRL=140,BT=C,RT=Z)
FILE(TAPE9,MRL=140,BT=C,RT=Z)
SORTMRG.
  (OTHER CONTROL CARDS TO PROCESS SORTED FILE TAPE9)
* 7/8/9      FOR
  (FORTRAN SOURCE CREATING TAPE7 ON DISK WITH FORMATTED WRITES)
* 7/8/9      EOR
  (FORTRAN DATA)
* 7/8/9      EOR
SORT
FILE,INPUT=TAPE7,OUTPUT=TAPE9
FIELD,MAJOR(23,3,DISPLAY),MINOR(2,15,DISPLAY)
KEY,MAJOR(A,COROL6),MINOR(D,COROL6)
END
" 6/7/8/9    EOF

```


JUNE 1977

PAGE 11-16

3. SORT USING SORT/MERGE 3.0 DIRECTIVES. SORT I/O FILES HAVE 40-WORD RECORDS WITH 12 RECORDS PER BLOCK.

```
JOBNAME,MT2,..... NAME / CODE
CHARGE,.....
LABEL,OLD,D=HY,L=SORTIN,VSN=CA9999,NORING.
LABEL,NEW,D=HY,L=SORTOUT,VSN=CA9998,RING.
FILE(OLD,BT=K,RT=F,FL=400,RE=12,MBL=4800)
FILE(NEW,BT=K,RT=F,FL=400,RE=12,MBL=4800)
LDSET(FILES=NEW/OLD)
SORTMRG(60)
" 7/8/9 FOR
(SORT/MERGE 3.0 DIRECTIVES)
" 6/7/8/9 EOF
```

*** GPSS (PROPRIETARY) ***

GENERAL PURPOSE SIMULATION SYSTEM (GPSS) V, VERSION 1.2, IS A GENERALIZED SIMULATION PACKAGE. IT IS COMPATIBLE WITH IBM SYSTEM/360 GPSS V. TO USE:

ATTACH,GPSS.
GPSS(<PARAMETERS>)

*** OMNITAB ***

OMNITAB II, OBTAINED FROM THE NATIONAL BUREAU OF STANDARDS, ENABLES THE NON-PROGRAMMER TO USE A LARGE DIGITAL COMPUTER TO PERFORM DATA, STATISTICAL AND NUMERICAL ANALYSIS WITHOUT HAVING ANY PRIOR KNOWLEDGE OF COMPUTERS OR COMPUTER PROGRAMMING.

JORNAME,CM170000,.... NAME / CODE
CHARGE,....
ATTACH,OMNITAB.
OMNITAB.
...

*** PL/I SUBSET ***

PL1, FROM THE COURANT INSTITUTE OF MATHEMATICAL SCIENCES, IS A SUPSET OF THE PL/I LANGUAGE AND IS ORIENTED TOWARD COMPILER WRITING. IT IS NOT COMPLETELY COMPATIBLE WITH IBM IMPLEMENTATIONS.

*** SCORE (PROPRIETARY) ***

SCORE IS A GENERALIZED RETRIEVAL AND REPORTING SYSTEM. IT IS A COBOL GENERATOR AND PROVIDES A SOURCE PROGRAM LISTING AS A BYPRODUCT OF THE USER SPECIFICATIONS. SCORE CAN THUS BE USED AS A REPORT WRITER, FILE GENERATOR AND/OR A SKELETON COBOL PROGRAM GENERATOR.

*** SPSS (PROPRIETARY) ***

STATISTICAL PACKAGE FOR THE SOCIAL SCIENCES (SPSS) IS AN OPEN-ENDED INTEGRATED SYSTEM OF STATISTICAL PROGRAMS EMBEDDED IN A SINGLE CONTROL PROGRAM. THE CDC 6000 VERSION WAS OBTAINED FROM NORTHWESTERN UNIVERSITY AND IS MAINTAINED BY ONE OF OUR CUSTOMERS (THE CONSTRUCTION ENGINEERING RESEARCH LABORATORY (CERL), COMPUTER SERVICES BRANCH). SPSS IS A BATCH SYSTEM WRITTEN MOSTLY IN FORTRAN. THIS PACKAGE (VERSION 6) IS MORE VERSATILE THAN THE BIMED ROUTINES (CCRM, 10-19), SINCE MANY DIFFERENT STATISTICS CAN BE PERFORMED ON THE SAME DATA IN ONE RUN. FOR LARGER AMOUNTS OF DATA, CM140000 SHOULD BE SPECIFIED ON THE JOB CARD; HOWEVER, THE SYSTEM WILL REDUCE CORE TO THE LOWEST NEEDED FOR EACH TYPE OF STATISTIC.

CURRENT PROGRAMS INCLUDE:

DESCRIPTIVE STATISTICS
 CROSS TABULATION
 PEARSON AND RANK ORDER CORRELATION
 PARTIAL CORRELATION
 MULTIPLE REGRESSION ANALYSIS
 NON-LINEAR REGRESSION ANALYSIS
 GUTTMAN SCALE ANALYSIS
 FACTOR ANALYSIS
 CANONICAL CORRELATION
 BIVARIATE PRINTER PLOTTING
 T-TEST
 ONE-WAY ANALYSIS OF VARIANCE
 N-WAY ANALYSIS OF VARIANCE
 DISCRIMINANT ANALYSIS
 NONPARAMETRIC STATISTICAL TESTS
 ITEM AND SCALE RELIABILITY ANALYSIS
 TETRACHORIC CORRELATIONS

EXAMPLE

XXXXYYY,CM70000. NAME/CODE
 CHARGE,XXXX,JJJJJJJJJJ.

 ATTACH,SPSS.

SPSS(<PARAM>) ** PARAM IS OPTIONAL PARAMETERS AS DESCRIBED IN
 ' 7/8/9 EOR APPENDIX F OF SPSS, 2ND EDITION

 " 6/7/8/9 EOF

REFERENCES:

"SPSS, SECOND EDITION", NIE, HULL, JENKINS,
 STEINBRENNER, BENT
 "SPSS PRIMER", KLECKA, NIE, HULL

*** SIMSCRIPT I.5 ***

SIMSCRIPT I.5 IS A PROGRAMMING LANGUAGE TO SIMULATE A REAL SITUATION WHICH CHANGES OVER SOME TIME INTERVAL. IT IS COMPATIBLE WITH OTHER SIMSCRIPT I.5 COMPILERS AND INCLUDES SOME DATA PROCESSING FEATURES AND HAS ACCESS TO FTN LIBRARY ROUTINES. USE THE FOLLOWING CONTROL CARDS:

```
ATTACH,SIMI5,ID=CSYS.  
SIMI5,<PARAMS>.  
LGO.
```

*** SIMSCRIPT II.5 ***

THE SIMSCRIPT II.5 COMPILER, WHICH IS NOT COMPATIBLE WITH SIMSCRIPT I.5, IS NOW AVAILABLE.

```
ATTACH,SIMII5,ID=CSYS.  
SIMII5,<OPTIONS>.  
ATTACH,SIM2LIB,ID=CSYS.  
LGO.
```

*** SNOBOL ***

SNOBOL IS A STRING MANIPULATION INTERPRETIVE PROGRAMMING LANGUAGE DESIGNED TO OPERATE UPON CHARACTER STRINGS RATHER THAN NUMERICAL DATA. SNOBOL DOES ACCEPT INTEGER AND REAL NUMERIC OPERATIONS, HOWEVER EXECUTION TIME IS GREATER THAN FOR A COMPILER SUCH AS FORTRAN. SNOBOL PROVIDES AN EXTENSIVE LIST OF FUNCTIONS TO OPERATE UPON STRINGS. PATTERNS USED FOR SUBSTRING MATCHING MAY BE SIMPLE OR COMPLICATED. SNOBOL IS A PUBLIC-ACCESS PERMANENT FILE.

DETAILED DESCRIPTION OF SNOBOL IS IN "THE SNOBOL4 PROGRAMMING LANGUAGE" BY R. E. GRISWOLD.

*** SYSTEM 2000 (PROPRIETARY) ***

SYSTEM 2000 IS A DATA BASE MANAGEMENT SYSTEM AVAILABLE TO ALL NAVY USERS OF THE DTNSRDC 6000 COMPUTERS. THE BASIC NATURAL LANGUAGE IS SUITABLE FOR INTERACTIVE USE. A PROCEDURAL LANGUAGE FEATURE ALLOWS MANIPULATION OF SYSTEM 2000 DATA BASES FROM COBOL OR FORTRAN PROGRAMS.

PROCEDURE 'S2000' MUST BE USED TO ACCESS THE SYSTEM 2000 FILES (SEE CCLIB/P).

*** VENUS ***

VENUS, VERSION 3.0, IS A RETRIEVAL SYSTEM DESIGNED FOR USE WITH INDEXED SEQUENTIAL (FO=IS) FILES. IT OPERATES IN THE INTERACTIVE OR BATCH MODE. FEATURES INCLUDE:

- .INTERACTIVE DATA BASE DEFINITION AND CREATION
- .INTERACTIVE UPDATE
- .BATCH MULTI-FILE CAPABILITY
- .PRINTER PLOT
- .INTERFACES TO USER PROGRAMS
- .DIRECT USAGE OF VENUS FILES BY HIGHER LEVEL LANGUAGES
- .SUB-FILES
- .UNLIMITED REDEFINITION OF FILES
- .TEXT SCANNING
- .REPORT WRITER
- .CALCULATOR MODE

SOME OF THE FEATURES (SUCH AS PLOT, REPORT WRITER) CAN BE USED WITHOUT VENUS.

VENUS WILL BUILD INDEXED SEQUENTIAL FILES FROM SEQUENTIAL INPUT. IT HAS BEEN MODIFIED TO ALLOW FOR RETRIEVAL FROM SEQUENTIAL FILES.

VENUS CURRENTLY REQUIRES LESS THAN CM55000.

REFERENCES: VENUS 3.0 USER MANUAL
VENUS SUPPORT PROCESSORS USER MANUAL

***** OTHER COMPUTER CENTER EQUIPMENT *****

THIS CHAPTER DESCRIBES OTHER COMPUTERS AND COMPUTER-RELATED HARDWARE AVAILABLE.

*** 1700 COMPUTER TERMINALS ***

THE CDC-6700 COMPUTER AT DTNSRDC CURRENTLY HAS THREE CDC-1700 COMPUTER TERMINALS. THE CDC-6600 COMPUTER AT DTNSRDC HAS ONE. THEY PROVIDE REMOTE/STAND-ALONE BATCH PROCESSING. THESE TERMINALS ARE:

| DESIGNATION | REMOTE COMPUTER | LOCATION | TERMINAL ID | CENTRAL COMPUTER |
|---------------|--------------------|-----------------|----------------|---------------------|
| DTNSRDC | SC-1700 | BLDG 192 | AD | 6700 (MFA) |
| DTNSRDC/ANNAP | 1700 | BLDG 100 | AE | 6600 (MFB) |
| NAVSEC | SC-1700 | NATIONAL CENTER | AF | 6700 (MFA) |
| NAVAIR | 1700 | JEFFERSON PLAZA | AG | 6700 (MFA) |

THE 1700 TERMINALS CAN OPERATE IN ONE OF THREE MODES:

1. STAND-ALONE: THE 1700 COMPUTER CAN COMPILE AND RUN 1700-FORTRAN OR 1700-ASSEMBLY JOBS.
2. TERMINAL: 6700 OR 6600 REMOTE PROCESSING.
3. STAND-ALONE/TERMINAL: CAN OPERATE ASYNCHRONOUSLY PROVIDING:
 - (A) CORE IS AVAILABLE FOR LOCAL JOBS;
 - (B) NO I/O DEVICE IS BEING SHARED BETWEEN THE TWO MODES OF OPERATION.

CAPABILITIES AVAILABLE

- | | | |
|---|-----------|------------|
| 1. REMOTE/STAND-ALONE JOB PROCESSING | | ALL |
| 2. IGS (GRAPHICS PROCESSING) | | AD, AF, AG |
| 3. PAPER TAPE (STAND-ALONE/REMOTE) | | AD, AE, AF |
| 4. CALCOMP PLOTTER (STAND-ALONE/REMOTE) | | AF, AG |
| 5. CARD LISTINGS, CARD DUPLICATION | | ALL |

DISPCAL_QE_DATA_ID_A-1700-EBOM-CENTRAL-SITE (SEE CCRM, 2-8, 14-3)

DISPCSE,LFN,X=IID.

OR

ROUTE,LFN,DC=X,TID=ID,FID=*

FOR IMMEDIATE ROUTING

ROUTE,LFN,DC=X,TID=ID,FID=*,DEF. FOR ROUTING AT END OF JOB

*** 1700 REFERENCE MANUALS ***

THE FOLLOWING CDC MANUALS ARE USEFUL FOR APPLICATION PROGRAMMERS:

| | PUB. NO |
|---|------------|
| MSOS VERSION 4 REFERENCE (1700 STAND ALONE OPERATING SYSTEM) | 60361500 E |
| 1700 MSOS IMPORT HIGH SPEED (6000/1700 REMOTE OPERATING PROCEDURE DESCRIPTION) | 60305700 A |
| MS FORTRAN VERSION 3A/B REFERENCE MANUAL (1700 FORTRAN) | 60362000 D |
| 1700 COMPUTER REFERENCE MANUAL (CHARACTERISTICS, OPERATION AND CAPABILITIES OF THE 1700 COMPUTER) | 60153100 |
| 1700 MSOS 4 MACRO ASSEMBLER REFERENCE MANUAL (SYMBOLIC LANGUAGE) | 60361900 C |
| 1700 MSOS 4 MACRO ASSEMBLER GENERAL INFORMATION | 39519800 B |

*** 200 USER TERMINAL ***

MEDIUM SPEED TERMINALS (ASCII OR BCD) OPERATE UNDER THE CONTROL OF INTERCOM AND CONSIST OF A CARD READER, A LINE PRINTER AND USUALLY A CRT DISPLAY. THE CDC 200 USER TERMINAL (200-UT) CAN BE CONNECTED DIRECTLY TO THE 6000 COMPUTERS. A NUMBER OF OTHER TERMINALS (INCLUDING MOHAWK, CDC 734 AND DATA 100) HAVE SOFTWARE PACKAGES (BOTH ASCII AND BCD) WHICH EMULATE THE 200-UT. ONE OF THE 200-UT'S (TERMINAL ID AB, IN ANNAPOLIS) IS HARDWIRED TO THE 6700. ALL OTHERS, INCLUDING 200-UT-EQUIVALENTS, ARE DIAL-UP. BOTH 2000 AND 4800 BAUD TELEPHONE LINES ARE AVAILABLE (SEE CCRM, 1-1,2).

A SET OF REMOTE BATCH PROCESSING COMMANDS (CCRM, 12-4) ALLOWS FOR READING AND PRINTING FILES. PRINT FILES MAY BE ROUTED TO A 200-UT FROM BATCH JOBS OR TTY. 200-UT OPERATORS MAY ALTER PRIORITY, DIVERT OR DROP JOBS IN THEIR QUEUES.

WHEN READING CARDS, A SWITCH ON THE 200-UT IS SET TO INDICATE 026 OR 029 PUNCH MODE. THEREFORE, THE ENTIRE DECK MUST BE PUNCHED IN ONLY ONE MODE. NO BINARY CARDS CAN BE READ. SINCE THE 200-UT DOES NOT HAVE A CARD PUNCH, USERS WHO WISH TO PUNCH CARDS MUST ROUTE THEM TO CENTRAL SITE OR TO A 1700.

WHEN PRINTING, THE ONLY CARRIAGE CONTROL CHARACTERS EFFECTIVE AT A 200-UT ARE:

| | |
|-------|--|
| BLANK | SINGLE SPACE BEFORE PRINTING |
| 1 | EJECT TO TOP OF PAGE BEFORE PRINTING |
| 0 | DOUBLE SPACE BEFORE PRINTING |
| + | SUPPRESS SPACING BEFORE PRINTING (PRINT ON SAME LINE) |
| PM | PRINTER MESSAGE |

ALL OTHERS ARE TREATED AS BLANKS.

THE 200-UT MAY BE USED ON LOCAL FOR CARD LISTING.

AT 4800 BAUD, A 200-UT WILL READ ABOUT 300 CARDS PER MINUTE AND PRINT ABOUT 300 LINES PER MINUTE. THEREFORE, CARD DECKS SHOULD BE LIMITED TO ONE BOX AND PRINTING TO 50 PAGES.

*** NOVA 800 ***

THE NOVA 800 IS A PROGRAMMABLE TERMINAL WHICH MAY BE USED AS A 200-UT EMULATOR FOR MEDIUM SPEED BATCH ACCESS TO CDC 6000 ON 2000 BAUD BCD LINES. THE 200-UT EMULATOR IS LOADED FROM A 9-TRACK TAPE (RT01/200UT).

*** REMOTE BATCH PROCESSING - MEDIUM SPEED ***

THE FOLLOWING COMMANDS ARE AVAILABLE AT A MEDIUM SPEED 200-UT-COMPATIBLE TERMINAL FOR PROCESSING BATCH JOBS AND OUTPUT FILES.

PRINTER

| | |
|------------------------------|--|
| ON,LP1 ON | TURN THE LINE PRINTER ON. MUST BE THE FIRST COMMAND AFTER LOGIN. |
| OFF,LP1 OFF | TURN THE PRINTER LOGICALLY OFF. ON,LP1 IS REQUIRED BEFORE PRINTING MAY RESUME. |
| END,LP1 E,LP1 END E | END PRINTING. DISCARD THE REMAINING OUTPUT. GO,LP1 TO PRINT THE JOB'S DAYFILE. IGNORE REPEAT COUNT, IF ANY. A SECOND END,LP1 WILL STOP PRINTING THE DAYFILE. |
| GO,LP1 G,LP1 GO G | RESUME PRINTING AFTER WAIT OR SOME ERROR CONDITIONS. |
| CONTIN C | RESUME PRINTING WHEN TRANSMISSION WAS STOPPED BECAUSE A MESSAGE WAS RECEIVED (AND IS DISPLAYED ON THE CRT) OR THE INTERRUPT KEY 'INTER' WAS PRESSED. |
| BSP,LP1,N BSP,,N BSP | BACKSPACE LINE PRINTER N (TIMES 10 OCTAL) SECTORS. (DEFAULT: 1 (10 OCTAL SECTORS)) |
| REP,LP1,M REP,,M REP | REPRINT M (OCTAL) ADDITIONAL COPIES (DEFAULT: 1; MAX: 37 OCTAL). THE REPEAT COUNT (RC) IS CUMULATIVE (I.E., REP,,2 FOLLOWED BY REP,,3 IS THE EQUIVALENT OF REP,,5). M MAY NOT BE NEGATIVE. |
| REW,LP1 REW | REWIND THE PRINT FILE AND TURN PRINTER LOGICALLY OFF. ON,LP1 IS REQUIRED BEFORE PRINTING MAY RESUME. THE RESIDUAL REPEAT COUNT IS SAVED. |
| RTN,LP1 RTN | REWIND THE PRINT FILE AND RETURN IT TO THE OUTPUT QUEUE. THE RESIDUAL REPEAT COUNT IS SAVED. |
| WAIT,LP1 WAIT | TEMPORARILY HALT PRINTING. TO RESUME, USE GO,LP1. |

CARD READER

ON,CR1 TURN CARD READER ON.

OFF,CR1 TURN CARD READER LOGICALLY OFF. ON,CR1 IS REQUIRED BEFORE
READING MAY RESUME.

END,CR1 END CARD READING.
E,CR1

GO,CR1 RESUME CARD READING AFTER WAIT OR SOME ERROR CONDITIONS.
G,CR1

CONTIN
C RESUME READING WHEN TRANSMISSION WAS STOPPED BECAUSE A
MESSAGE WAS RECEIVED (AND IS DISPLAYED ON THE CRT) OR THE
INTERRUPT KEY 'INTER' WAS PRESSED.

READ
R READ CARDS INTO THE INPUT FILE FOR THIS TERMINAL.

WAIT,CR1 TEMPORARILY HALT CARD READING. TO RESUME, USE GO,CR1.

PRINTER AND CARD READER

WAIT,ALL TEMPORARILY HALT READING AND PRINTING. TO RESUME, USE GO.

GO,ALL RESUME READING AND/OR PRINTING WHICH WAS HALTED BY WAIT.
G,ALL ALSO REQUIRED, WITH OR WITHOUT EQUIPMENT, WHEN ANY OF THE
FOLLOWING ERRORS HAVE OCCURRED:
 DEVICE NOT READY
 JOB STATEMENT ERROR
 INPUT FILE ERROR
 OUTPUT FILE ERROR

CONTIN
C RESUME READING AND/OR PRINTING WHEN TRANSMISSION WAS
STOPPED BECAUSE A MESSAGE WAS RECEIVED (AND IS DISPLAYED
ON THE CRT) OR THE INTERRUPT KEY 'INTER' WAS PRESSED.

QUEUE AND STATUS DISPLAYS

H,I DISPLAY INPUT QUEUE FOR THIS TERMINAL.

H,O DISPLAY OUTPUT QUEUE FOR THIS TERMINAL.

H,E DISPLAY JOBS IN EXECUTION FOR THIS TERMINAL.

H,S DISPLAY CURRENT STATUS OF OF THIS TERMINAL'S I/O DEVICES.

H,P DISPLAY PUNCH QUEUE FOR THIS TERMINAL. (JOBS IN THE PUNCH
QUEUE MUST BE DIVERTED TO ANOTHER TERMINAL FOR PUNCHING.
SEE BELOW.)

MANIPULATING QUEUE-RESIDENT FILES

IN THE FOLLOWING COMMANDS, JOBNAME IS THE NAME OF THE JOB/FILE AFFECTED. IT MAY BE THE 7-CHARACTER JOB NAME OR THE LAST 2 CHARACTERS OF THE NOS/BE JOB NAME.

DIVERT, JOBNAME, ID, Q

TRANSFER FILES/JOB'S TO ANOTHER LOCATION.

JOBNAME - NAME OF JOB TO BE TRANSFERRED.

ID - ID OF TERMINAL TO RECEIVE DIVERTED JOB. IF OMITTED, DIVERT TO CENTRAL SITE.

Q - DIVERT JOBS FROM THIS QUEUE ONLY. Q MAY BE O (PRINT), P (PUNCH), I (INPUT), OR OMITTED (ALL QUEUES).

EXAMPLES: DIVERT, CXXX0R7, PA, O

DIVERT PRINT FILE

CXXX0R7 TO TERMINAL PA.

DIVERT, R7, , P

DIVERT PUNCHING OF JOB R7
TO CENTRAL SITE.

DROP, JOBNAME

DROP A JOB WHILE IT IS IN EXECUTION. ANY OUTPUT IS PLACED INTO THE OUTPUT QUEUE.

EVICT, JOBNAME, Q

ELIMINATE JOB FROM SPECIFIED QUEUE. Q MAY BE I (INPUT), O (PRINT), P (PUNCH), OR OMITTED (ALL QUEUES).

PRIOR, JOBNAME, P, Q

CHANGE PRIORITY OF A PRINT OR PUNCH FILE. P IS A 1-4 DIGIT POSITIVE OCTAL NUMBER DEFINING THE NEW PRIORITY. JOBS WITH PRIORITY 0000 WILL REMAIN IN THE QUEUE. Q MAY BE O (PRINT), P (PUNCH) OR OMITTED (PRINT AND PUNCH).

SOME ERROR MESSAGES

COMMAND IGNORED

COMMAND CANNOT BE PROCESSED. (E.G., OFF,LP1 CANNOT BE EXECUTED WHEN THE PRINTER IS ALREADY OFF.)

COMMAND REJECT

EITHER GO OR ON COMMAND MUST BE ENTERED BEFORE OTHER COMMANDS MAY BE PROCESSED.

CRN,JJJJJJJ,INPUT FILE ERROR

ERRORS IN CREATING INPUT FILE. JOB JJJJJJJJ IS DROPPED FROM THE SYSTEM. CHECK CARDS FOR ERRORS. END,CR AND TRY AGAIN.

CRN,JJJJJJJ,SHIFTED DATA,INPUT FILE ERROR

A CARD WAS READ INCORRECTLY. TO PROCEED: END,CR1; CHECK FOR A DAMAGED CARD IN THE LAST GROUP READ AND FIX; PUT AN EOF CARD (6/7/8/9) IN FRONT OF THE JOB CARD (FOR SAFETY) AND GO,CR1 TO READ COMPLETE DECK.

CRN,JJJJJJJJJJ,JOB CARD ERROR

JOB CARD IS INCORRECT. JJJJJJJJJJ IS THE JOB NAME OR THE FIRST 10 CHARACTERS OF THE JOB CARD. RETURN TO USER FOR CORRECTION.

CRN,NOT READY

PLACE MORE CARDS (OR AN END-OF-FILE CARD) IN READER. PRESS LOAD. ENTER GO,CRN.

INPUT SUSPENDED BY SYSTEM

SYSTEM INPUT QUEUE IS OVERLOADED. READING RESUMES AUTOMATICALLY WHEN OVERLOAD CONDITION ENDS. (PRINTER MUST BE LOGICALLY ON.)

LPN,NOT READY

CORRECT CONDITION. READY PRINTER. ENTER GO,LPN.

*** STROMBERG DATAGRAPHIX SC4060 ***

THE STROMBERG DATAGRAPHIX SC4060 IS A STORED PROGRAM RECORDING SYSTEM WHICH TRANSLATES DIGITAL DATA INTO ALPHANUMERIC AND GRAPHIC DATA. INPUT IS FROM 7-TRACK TAPE WHICH SHOULD BE 556 BPI DENSITY. OUTPUT IS 35 MM FILM WHICH MAY OPTIONALLY BE PRINTED ONTO ROLL PAPER BY XEROX COPYFLOW. THE DEFAULT OUTPUT IS PAPER CUT INTO SHEETS AT CUT MARKS INSERTED ON THE FILM FROM A FORM SLIDE AT EACH FRAME ADVANCE. THE QUALITY AND RESOLUTION ARE BETTER THAN SC4020. PRINT STYLE IS DIFFERENT SINCE LETTERS HAVE SERIES.

SC4020 STRANGER TAPES WRITTEN BY SCORS PACKAGE ON CDC ARE PROCESSED BY PROGRAM "4020". OUTPUT IS SIMILAR TO SC4020 EXCEPT FOR SCALE AND PULL DOWN BETWEEN FRAMES, PLOTS ARE LARGER AND SPACE BETWEEN FRAMES IS GREATER.

SYSTEM STANDARD PRINT TAPES MAY BE PRINTED ONTO 35 MM FILM USING SC4060 PROCESS PROGRAM "PCDC". ARCHIVE RECORDS COULD BE PREPARED ON FILM FOR COMPACT STORAGE. PRINT TAPES FROM SEVERAL OTHER COMPUTER SYSTEMS MAY BE PROCESSED DIRECTLY TO FILM USING SUITABLE PROCESSING PROGRAMS SUCH AS "P709" FOR IBM 7090 COMPATIBLE PRINT TAPES.

A PACKAGE OF ROUTINES WHICH OUTPUT "META" CODE EXCLUSIVELY FOR SC4060 IS MORE DIFFICULT TO LEARN THAN SCORS ROUTINES, BUT HAS MANY ADDITIONAL OPTIONS. THE RASTER SIZE OF THE SC4060 IS 3072 X 4096. THE 'DISSPLA' PACKAGE HAS 'META' OUTPUT FOR SC4060.

THE ADP OFF-LINE REQUEST MUST INCLUDE THE NAME OF THE PROCESS PROGRAM TO BE USED:

| | |
|------|--|
| 4020 | FOR SC4020 STRANGER TAPES WRITTEN BY SCORS PACKAGE |
| PCDC | CDC STANDARD PRINT TAPES |
| META | SC4060 INTEGRATED GRAPHICS SOFTWARE |

TO ORDER FILM ONLY WITHOUT CUT MARKS BETWEEN FRAMES OR ROLL PAPER THAT IS NOT CUT INTO SHEETS, PUT "NO CUT MARKS" ON ADP REQUEST.

*** COM - DATAGRAPHIX 4530 ***

COMPUTER OUTPUT MICROFICHE (COM) IS PREPARED ON THE STROMBERG DATAGRAPHIX 4530. INPUT IS 7-TRACK TAPE AT 800 OR 556 BPI, OR 9-TRACK TAPE AT 800 BPI. INPUT TAPES ARE PREPARED FROM FORMATTED OUTPUT OR REPORT FILES BY THE MPS (MICROMATION PRINTER SYSTEM) PACKAGE.

OUTPUT IS 105MM FILM. EACH 4 BY 6 INCH (105MM BY 148MM) MICROFICHE MAY CONTAIN 268 COMPLETE PAGES PLUS EYE-READABLE TITLE AND INDEX AT STANDARD NMA 48X REDUCTION RATIO. PAGE FORMATTING ALLOWS 80, 132 OR 144 CHARACTERS PER LINE AND UP TO 80 LINES PER PAGE. 5 BY 6 INCH CUT FILM NEGATIVES MAY BE CREATED FOR FORMS OVERLAYS. FICHE FRAMES ARE IN COLUMNS (WIDTH) AND ROWS (LENGTH) NUMBERED FROM A01 TO 018 FOR EASY REFERENCE.

SEVERAL JOB SET UP CARDS ARE AVAILABLE ON THE 4530. PRINT SPEED IS APPROXIMATELY 14000 LINES PER MINUTE.

THE MPS PACKAGE IS CAPABLE OF PRODUCING FICHE WITH ALL CDC PRINTABLE CHARACTERS EXCEPT THE BACK SLASH.

A COM TAPE MAY BE PREPARED AS FOLLOWS:

```
XXXX,CM55000,MT1.  
CHARGE,XXXX,JJJJJJJJJJ.  
ATTACH,TAPE03,.... (FILE TO BE PUT ONTO MICROFICHE)  
LABEL,TAPE04,L=XXXXCOMOUT,D=HY,F=S,RING,W,VSN=CB9999.  
BEGIN,UTILITY,,MPSCM.  
" 7/8/9 EOR  
  (MPS DATA CARDS)  
" 6/7/8/9 EOF
```

REFERENCE: "COMPUTER OUTPUT MICROFICHE REFERENCE MANUAL", CMD-16-76,
BARBARA DAVIS

*** CALCOMP MODEL 936 ***

THE CMLD CALCOMP MODEL 936 DRUM PLOTTER IS SOURCE CODE COMPATIBLE WITH CALCOMP 736.

THE FOLLOWING PLOTTING PAPERS AND PENS WILL BE AVAILABLE TO USERS:

1. 36.7 INCH PLAIN PAPER #500
2. 36.7 INCH 10 DIVISIONS PER INCH (BROWN GRID) #501
3. 15.4 INCH PLAIN PAPER #600
4. 15.4 INCH 10 DIVISIONS PER INCH (BROWN GRID) #601
5. 3 BALL POINT PENS
A CALL TO SUBROUTINE NEWPEN (N), WHERE N = 1, 2, OR 3 WILL ALLOW USERS TO ALTERNATE 3 PENS IN ONE JOB. COLORS GREEN, RED, BLUE AND BLACK ARE AVAILABLE. SEE CALCOMP BASIC MANUAL FOR DESCRIPTION OF NEWPEN.

THE BASIC SET-UP FOR THE CALCOMP 936 WILL BE AS FOLLOWS:

- * 15.4 INCH PAPER (#600 OR #601)
- * PEN 1 (BLACK)
- * PEN 2 (BLUE)
- * PEN 3 (RED)

A CHANGE TO WIDE PAPER (36.7 INCH) OR A CHANGE IN PEN ARRANGEMENT WILL INCREASE THE BASIC CHARGE RATE.

THE ABILITY TO DISPLAY AND SEARCH FOR BLOCK ADDRESS IS NOT AVAILABLE ON THE CALCOMP 936. USERS WHO NEED TO SELECT BLOCKS FROM THEIR CALCOMP TAPE MUST USE A CDC 6000 PROGRAM.

*** CALCOMP MODEL 763 ***

CALCOMP MODEL 763 IS A DRUM TYPE DIGITAL ZIP MODE PLOTTER WITH A PLOTTING AREA OF 29.5 INCHES BY 120 FEET, A CHOICE OF THREE STEP SIZES, AND AN INCREMENTAL RATE OF 350 OR 450 STEPS PER SECOND DEPENDING ON STEP SIZE. THE MAGNITUDE OF THE STEP SIZE IS .01 OR .005 INCH WITH A RESULTING ABILITY TO PRODUCE INCREMENTAL VECTORS IN 24 DIFFERENT COMBINATIONS OF LENGTH.

THE PLOTTER WHICH IS LOCATED IN BUILDING 13, OPERATES OFF-LINE FROM HI DENSITY (556 BPI) 7-TRACK TAPE. THE AERO COMPUTER CENTER PHONE IS (202) 227-1423. TAPES MAY BE LEFT AT ADP CONTROL WITH PROPER FORMS AND THE PLOTS SHOULD BE RETURNED WITH ONE DAY TURN AROUND.

SUBROUTINE CALLS ARE USED ON CDC 6000 OR OTHER COMPUTER SYSTEMS TO GENERATE THE TAPE. PAGE 10-18 COVERS CDC 6000 REQUIREMENTS AND "PROGRAMMING CALCOMP PEN PLOTTERS" DESCRIBES THE SIX BASIC ROUTINES. SEVERAL OTHER PACKAGES OF CALCOMP ROUTINES HAVE BEEN PURCHASED.

THE FEATURES OF THE SYSTEM INCLUDE ABILITY TO DISPLAY AND SEARCH FOR BLOCK ADDRESS, MAXIMUM INCREMENTAL SPEED 6.3 IN/SEC, MAX ZIP MODE SPEED 23.8 IN/SEC, MAX INCHES OF PLOT PER INCH OF INPUT MAGNETIC TAPE 14.7, INTERCHANGE OF AXES AND PEN CONTROL BY SWITCH, NO PARITY CHECK SYSTEM, FAST FORWARD AND REWIND.

*** CALCOMP PLOT REQUEST ***

PLOT REQUEST (FORM 10462/26) MUST BE SUBMITTED WITH EACH TAPE. USER MUST PROVIDE PERTINENT INFORMATION FOR THE FIRST TWO SECTIONS FROM LEFT INCLUDING JOB ORDER NUMBER, ESTIMATED PLOTTING TIME TO AID IN SCHEDULING OF WORK, DATE, BLOCK NUMBERS OF REEL IF KNOWN (USUALLY 0 TO 999), REEL NUMBER, EXPECTED NUMBER OF PLOTS, PAPER NUMBER AND GRID COLOR FROM NEXT PAGE, PEN TYPE (BALL POINT GIVES BEST RESULTS), USER ORIGIN IF NONSTANDARD, USER NAME, PHONE NUMBER.

*** 763 CALCOMP PAPER AND GRID COLOR ***

| PAPER NUMBER | DESCRIPTION |
|--------------|--|
| 300 BROWN | PLAIN 30 IN WIDE PAPER |
| 301 BLUE | CONTINUOUS GRID (10 PER INCH) 30 INCH WIDE |
| 302 BLUE | CONTINUOUS GRID (20 PER INCH) 30 INCH WIDE |
| 400 BROWN | PLAIN 12 INCH WIDE PAPER |
| 400-A BROWN | PAGE SIZE (8.5 X 11) PLAIN |
| 401 BLUE | CONTINUOUS GRID (10 PER INCH) 12 INCH WIDE |
| 401 BROWN | CONTINUOUS GRID (10 PER INCH) 12 INCH WIDE |
| 401-A BLUE | PAGE SIZE (7 X 10) BLUE GRID (TEAR OFF) 10 PER INCH |
| 401-A BROWN | PAGE SIZE (7 X 10) BROWN GRID (TEAR OFF) 10 PER INCH |
| 401-B BLUE | DOUBLE PAGE SIZE (15 X 10) BLUE GRID |
| 401-B BROWN | DOUBLE PAGE SIZE (15 X 10) BROWN GRID |
| 402 BLUE | CONTINUOUS GRID (20 PER INCH) 12 INCH WIDE |
| 405 BROWN | CONTINUOUS GRID CENTIMETER (10 PER CM.) 25 CM. WIDE |
| 405-A BROWN | PAGE SIZE (19 CM. X 25 CM.) CENTIMETER (TEAR OFF) |
| 405-B BROWN | DOUBLE PAGE SIZE (38 CM. X 25 CM.) TEAR OFF |

*** GSA-ATLANTA TIME SHARING SERVICE ***

THE GSA TIME SHARING SYSTEM IS AN ALTERNATIVE SYSTEM TO INTERCOM WHICH USES A GE440 COMPUTER WITH 64,000 (24-BIT, 3-CHARACTER) WORDS OF MEMORY. EACH USER HAS AVAILABLE 16,000 WORDS (WHICH CAN BE INCREASED TO 18,500 BY USING THE COMMAND 'RUNBIG'). TERMINAL ACCESS IS 110 OR 300 BAUD, HALF DUPLEX, TTY COMPATIBLE. PROGRAMMING LANGUAGES ARE FORTRAN AND BASIC. EDIT AND SAM ALLOW FILE MANIPULATION SUCH AS RESEQUENCE, DELETE, MERGE, AND REPLACE. CDC 6000 INTERCOM SHOULD BE USED EXCEPT WHEN EXTENDED BASIC OR SPECIAL LIBRARY ROUTINES ARE REQUIRED.

OPERATING HOURS ARE 0800 - 2300 WEEKDAYS
1000 - 1700 SATURDAYS FEDERAL HOLIDAYS EXCEPTED.

ACCESS TO THE SYSTEM IS OBTAINED BY FOLLOWING THESE STEPS:

1. REGISTER CHARGEABLE JOB ORDER NUMBER WITH CODE 1891, (202) 227-1910. YOU WILL BE ASSIGNED A USER ID AND PASSWORD.
2. DEPRESS ORIG BUTTON AND DIAL 9-755-2000 (FROM MODEL 33 OR 35 TTY OR OTHER 110-BAUD TERMINAL) OR 9-755-2010 (FROM 300-BAUD TERMINAL).
THE COMPUTER RESPONDS: GSA 440 TIME SHARING
ON AT XX:XX PN:XX TTY:XX
USER ID --
3. TYPE IN USER ID AND CARRIAGE RETURN (CR).
THE COMPUTER RESPONDS: PASSWORD
4. TYPE IN PASSWORD (CR).
THE COMPUTER RESPONDS: TYPE OLD OR NEW:
5. TO ENTER A NEW PROGRAM FROM KEYBOARD OR PAPER TAPE, TYPE 'NEW' TO CALL UP A PROGRAM FROM USER PRIVATE LIBRARY, TYPE 'OLD', THEN (CR).
THE COMPUTER RESPONDS: PROGRAM NAME.
6. FOR A NEW PROGRAM TYPE IN UP TO SIX CHARACTER NAME (CR).
FOR AN OLD PROGRAM, TYPE THE NAME UNDER WHICH THE PROGRAM WAS SAVED.
IF A LIBRARY PROGRAM, TYPE THE LIBRARY NAME FOLLOWED BY *** (CR).
(E.G. GAUSS1*** (CR))

GSA TIME SHARE MANUALS INCLUDE: A LIST OF LIBRARY ROUTINES, THE TERMINAL USERS GUIDE, AND REFERENCE MANUALS ON THE AVAILABLE LANGUAGES. THEY ARE LOCATED WITH EACH COORDINATOR OR THE USER SERVICES BRANCH, (202) 227-1907. A VOLUME OF LIBRARY PROGRAM LISTINGS IS AVAILABLE FOR INSPECTION IN CODE 1892. COORDINATORS ARE LISTED BELOW:

| NAME | BLDG | ROOM | EXT | CODE |
|---------------|------|------|-------|----------------------|
| G. MABRY | 3 | 259 | 71438 | 19 |
| J. NIEMIEC | 3 | 112B | 71636 | 1945 |
| D. CHADDOCK | 13 | 202A | 71670 | 16 |
| R. SENDELBACH | 15 | 100 | 71379 | 1936 |
| L. MUELLER | 17 | 100 | 71907 | DTNSRDC CO-ORDINATOR |
| J. DISKIN | 16 | 108 | 71450 | 15 |
| F. MEYER | 19 | B204 | 71751 | 17 |
| D. SOMMER | 100 | 2-J | 3343 | ANNAPOLIS |

*** AUTOMATED DRAFTING AND GRAPHIC DATA REDUCTION STATION ***

*** ALTEK AC-90 DIGITIZING SYSTEM ***

THE ALTEK AC-90 DIGITIZING SYSTEM IS THE FIRST PART OF A FACILITY REFERRED TO AS AN "AUTOMATED DRAFTING AND GRAPHIC DATA REDUCTION STATION". THE CURRENT CONFIGURATION INCLUDES A 42" X 54" BENDIX DIGITIZING TABLE WITH A FREE-MOVING CURSOR, A 52-CHARACTER KEYBOARD, A MAGNETIC TAPE DRIVE (7-TRACK, 800 BPI - COMPATIBLE WITH CDC 6000 SERIES COMPUTERS) AND IS ALSO INTERFACED TO AN 026 KEYPUNCH. RESOLUTION IS 0.001" WITH AN ACCURACY OF ± 0.005 " ACROSS THE BOARD. THE PRIMARY FUNCTION OF THE DIGITIZER IS TO CONVERT POSITIONAL GRAPHIC INFORMATION TO A USABLE FORM IN A COMPUTER DATA PROCESSING FORMAT. THE DIGITIZER MAY BE EQUATED TO A UNIQUE ANALOG-TO-DIGITAL CONVERTER WHEREIN THE ANALOG DATA IS PRESENTED IN A GRAPHIC FORM. IN ADDITION TO OUTPUTTING A SEQUENTIAL STREAM OF X,Y CO-ORDINATES OF POSITIONS SELECTED BY MOVEMENT OF THE CURSOR, THE DIGITIZER IS FITTED WITH A PORTABLE KEYBOARD TO PERMIT INSERTION OF SPECIAL CHARACTERS OR ALPHANUMERICS WITHIN THE STREAM. THESE CHARACTERS WOULD BE PLACED BY THE OPERATOR INTO THE STREAM TO COMMUNICATE INSTRUCTIONS TO SOFTWARE WHEN THE DATA IS PRESENTED TO THE COMPUTER. SOME BASIC SOFTWARE WAS ACQUIRED WITH THE DIGITIZER BUT ALL SPECIAL PURPOSE PROGRAMMING MUST BE USER-DEVELOPED. FOR ADDITIONAL INFORMATION, SEE CODE 1892.

*** GRAPHICS DATA REDUCTION AND DESIGN STATION ***

A GENERAL PURPOSE GRAPHICS DATA REDUCTION AND DIGITIZING STATION (GDRDS) CONSISTING OF A DIGITIZING BOARD, MINI-COMPUTER AND A CRT SCOPE HAS BEEN INSTALLED IN BUILDING 192. DIGITAL DATA DERIVED FROM DRAWINGS MAY BE DISPLAYED ON THE CRT FOR EDITING AND MAY THEN BE TRANSMITTED TO THE HOST COMPUTER, THE CDC 6000, FOR FURTHER PROCESSING.

THE GDRDS MAY BE USED EITHER AS A REMOTE GRAPHICS TERMINAL OR AS A STAND ALONE GRAPHICS DESIGN STATION. IT CAN BE EFFECTIVELY USED IN THE FOLLOWING APPLICATION AREAS:

- COMPUTER-AIDED PIPING DESIGN AND CONSTRUCTION
- SHIP HULL-FORM DESIGN
- PREPARATION OF INPUT DATA FOR POTENTIAL FLOW COMPUTATIONS
- PREPARATION OF INPUT DATA FOR STRUCTURAL ANALYSIS COMPUTATION

REFERENCE: "GRAPHICS DATA REDUCTION AND DESIGN STATION USER'S MANUAL",
CMD-11-76

ASSISTANCE: RUEY CHEN (202) 227-1683

*** REMOTE JOB ENTRY TO TI-ASC ***

DTNSRDC OPERATES A TEXAS INSTRUMENTS (TI) 980 REMOTE BATCH TERMINAL TO THE TI ADVANCED SCIENTIFIC COMPUTER (ASC). IT HAS A CARD READER, CARD PUNCH, LINE PRINTER, NINE-TRACK TAPE DRIVE, AND AN 11-INCH CALCOMP 565 DRUM PLOTTER.

THE TI-ASC IS A HIGH-SPEED SCIENTIFIC COMPUTER LOCATED AT THE NAVAL RESEARCH LABORATORY. IT IS BATCH-ORIENTED WITH TELETYPE ACCESS THROUGH A FRONT-END MINICOMPUTER. THE ASC ARITHMETIC UNIT IS A PIPELINE PROCESSOR (2 PIPES) WHICH ALLOWS IT TO PERFORM VECTOR ARITHMETIC AT VERY HIGH SPEEDS. APPROXIMATELY 700,000 32-BIT WORDS OF CENTRAL MEMORY AND 30 MILLION WORDS OF SECONDARY STORAGE ARE AVAILABLE TO USER PROGRAMS. THESE MAXIMUMS ARE EXPECTED TO INCREASE IN THE NEAR FUTURE.

THE JOB SPECIFICATION LANGUAGE (JSL) IS SIMILAR TO IBM JOB CONTROL LANGUAGE (JCL). STANDARD SOFTWARE IS SUPPLIED, INCLUDING AN OPTIMIZING FORTRAN COMPILER WHICH RECOGNIZES MANY LOOPS WHICH MAY BE EXECUTED IN VECTOR MODE. ADDITIONALLY, CERTAIN SOFTWARE TOOLS HAVE BEEN IMPLEMENTED ON THE ASC, INCLUDING RATFOR (CCRM, 4-28) AND EDIT (CCRM, 8-4). THE TAPE-COPY FACILITY PERMITS A USER TO USE THE ASC AS A "NUMBER CRUNCHER" AND THEN TRANSFER DATA TO DRIVE INTERACTIVE GRAPHICS PROGRAMS AVAILABLE AT DTNSRDC. THIS APPROACH IS ATTRACTIVE TO MANY USERS AND HAS PROVEN ESPECIALLY EFFECTIVE FOR PERFORMING INTRICATE FLUID DYNAMICS CALCULATIONS.

POTENTIAL USERS MAY CONTACT:

PAUL MORAWSKI, DTNSRDC CODE 1843, (202) 227-1932
BARBARA BROOKS, NRL, (202) 767-3542

*** NAVY LABORATORY COMPUTER NETWORK (NALCON) ***

THE NAVY IS CONDUCTING A THREE YEAR EXPERIMENT (FY76 THRU FY78) TO DETERMINE THE EFFECTIVENESS OF COMPUTER NETWORKING IN PROVIDING COMPUTING SUPPORT TO THE NAVY LABORATORIES.

IN SUPPORT OF THESE EXPERIMENTS, THE NAVY LABORATORY COMPUTER NETWORK (NALCON) WILL PROVIDE A FULL RANGE OF COMPUTING SERVICES TO ALL THE NAVY LABORATORIES AND, EVENTUALLY, TO ALL APPROPRIATE NAVY ACTIVITIES DESIRING ACCESS TO THE LABORATORY COMPUTERS. THE INITIAL SERVICE PROVIDED WILL BE INTERACTIVE PROCESSING. REMOTE JOB ENTRY, AND FILE TRANSFER, AS WELL AS PROCESS-TO-PROCESS COMMUNICATION WILL BE ADDED IN THE FUTURE. THE IMPLEMENTATION WILL INSULATE THE HOST COMPUTERS (UNIVAC 1100'S, CONTROL DATA CORPORATION 6000'S, TEXAS INSTRUMENTS-ASC, BURROUGHS 5500) FROM THE NETWORK ARCHITECTURE BY MEANS OF A MINICOMPUTER-BASED FRONT END SYSTEM. THE FRONT END WILL COMMUNICATE WITH THE HOST THROUGH A GENERAL HOST-FRONT END PROTOCOL AND WITH THE ARPANET THROUGH THE VERY DISTANT HOST INTERFACE.

ACCESS TO COMPUTING RESOURCES AT THE NAVY LABORATORIES PARTICIPATING IN NALCON AND AT OTHER ARPANET HOST COMPUTERS IS NOW AVAILABLE THROUGH THE NETWORK ACCESS MACHINE OPERATED BY DTNSRDC. RESOURCES THAT MAY BE OF INTEREST INCLUDE:

MACSYMA A SYMBOLIC MATHEMATICAL LANGUAGE AVAILABLE AT MIT.

MAIL ELECTRONIC MAIL SERVICE IS AVAILABLE OVER THE ARPANET ON A LIMITED BASIS.

NLS A SET OF COMPUTER-BASED AIDS FOR PEOPLE WHO WORK WITH INFORMATION. SERVICES AVAILABLE INCLUDE TEXT PROCESSING, MESSAGE HANDLING, DOCUMENT PRODUCTION, PHOTOCOMPOSITION, AND INFORMATION STORAGE, MANAGEMENT AND RETRIEVAL.

ANYONE NOW ACCESSING THE ARPANET THROUGH A REMOTE TIP CAN GET SERVICE THROUGH THE LOCAL NETWORK ACCESS MACHINE. IT WILL BE AVAILABLE SEVEN DAYS A WEEK, 18 HOURS A DAY.

FOR FURTHER INFORMATION ABOUT NETWORK RESOURCES AND SERVICES, CONTACT:

FRANK BRIGNOLI, DTNSRDC CODE 1826, (202) 227-1683

JUNE 1977

PAGE 12-17

*** FARRINGTON OCR ***

OPTICAL CHARACTER READING OF SPECIAL DOCUMENT FORMS IS PERFORMED ON THE FARRINGTON PAGE READER, MODEL 3030. THE OCR HAS A VARIAN DATA 620/I COMPUTER; OUTPUT IS ON MAGNETIC TAPE. A TELETYPE AND LINE PRINTER ARE PART OF THE SYSTEM.

***** USER INTERFACE *****

*** GENERAL INFORMATION ***

CONSULTATION IS AVAILABLE FROM THE USER SERVICES GROUP:

CARDEROCK: BLDG 17, ROOM 100, (202) 227-1907

ANNAPOLIS: BLDG 100, ROOM 2J, (301) 267-3343

FOR INFORMATION ABOUT:

SERVICES AVAILABLE - CODE 1892.1 (202) 227-1907

ACCOUNTING - CODE 1891 (202) 227-1361

HARDWARE - CODE 1894 (202) 227-1400

THE ADP CONTROL CENTERS ARE LOCATED AT THE CENTRAL SITE OF EACH CDC 6000. USERS MAY SUBMIT DECKS AND PICK UP OUTPUT AS WELL AS OBTAIN INFORMATION ON THE PROGRESS OF THEIR JOBS FROM THE ADP CONTROL CENTER.

THE COMPUTER USERS FORUM IS A GROUP OF REPRESENTATIVE USERS OF THE DTNSRDC COMPUTER FACILITIES. THE FORUM MEETS QUARTERLY TO EXCHANGE INFORMATION BETWEEN THE USERS AND THE COMPUTER CENTER MANAGEMENT. EACH DTNSRDC DEPARTMENT AND MAJOR OUTSIDE USER HAS REPRESENTATIVES ON THE FORUM. FOR MORE INFORMATION, CONTACT THE CHAIRMAN (USER SERVICES GROUP).

COMPUTER CENTER NOTES IS A PUBLICATION SENT TO ALL REGISTERED USERS. IT IS SENT WHENEVER THERE IS INFORMATION TO BE DISSEMINATED. THE SYSTEM BULLETIN PROVIDES CURRENT NEWS AND IS PRINTED AT THE BEGINNING OF EACH JOB AND EACH INTERCOM LOGIN (EXCEPT SUP).

OUTSIDE COMPUTER USERS WORK AREA IS IN TRAILER T-7 (EAST OF BLDG. 17), PHONE 227-1382.

*** USER TROUBLE FORM ***

A USER TROUBLE FORM (UTF) IS USED:

1) FOR REFUND REQUESTS (SEE CCRM, 2-1; POLICY, 13)

2) WHEN PROBLEMS ARE ENCOUNTERED;

3) FOR SUGGESTIONS, GRIPEs AND COMPLAINTS.

THE UTF SHOULD INCLUDE A SUCCINCT DESCRIPTION OF THE PROBLEM AND INCLUDE AS MUCH DOCUMENTATION (DAYFILE, LISTINGS, DUMPS) AS POSSIBLE. IT SHOULD BE SUBMITTED TO CODE 1892 FOR PROCESSING.

TROUBLE FORMS WHICH DO NOT REQUIRE DOCUMENTATION MAY BE ENTERED DIRECTLY INTO THE COMPUTER FROM EITHER BATCH OR INTERCOM. THE PROCEDURE 'GRIPE' (CCRM, 7-22; CCLIB/P) IS USED FOR THIS.

*** GRIPEs ***

GRIPEs AND COMMENTS MAY BE SENT TO USER SERVICES VIA INTERCOM. USE 'BEGIN,SEND,,SEND,U' (SEE CCRM, 9-12; CCLIB/P: SEND). THE 'GRIPE' PROCEDURE (CCRM, 7-22) MAY ALSO BE USED (BATCH OR INTERCOM). ALL MESSAGES AND GRIPEs WILL BE READ AND TROUBLE FORMS PREPARED, IF APPROPRIATE.

*** COMPUTER REFERENCE CENTERS ***

A COMPLETE LIBRARY OF ALL CDC 6000 AND MANY OTHER COMPUTER PRODUCT DOCUMENTS (SEE CCRM, 13-3) MAY BE CONSULTED IN CODE 1892.1 (BLDG. 17, ROOM 100). SUBSETS OF FREQUENTLY USED MANUALS ARE AVAILABLE FOR REFERENCE AT A NUMBER OF OTHER SITES. THESE COMPUTER REFERENCE CENTERS ARE LISTED BELOW:

| CODE | NAME | LOCATION |
|--------|-----------------|-----------------------------|
| 152 | R. WERMTER | BLDG 2, RM 201 |
| 1544 | E. CASTER | BLDG 16, RM 105 |
| 1576 | W. MEYER | BLDG 18, RM 101 |
| 1681 | E. ROGERS | BLDG 7, RM 118 |
| 1725 | P. ROTH | BLDG 19, RM A248 |
| 1828 | A. CAMARA | BLDG 191, RM 121 |
| 1843 | R. VAN ESELTINE | BLDG 192, RM 128 |
| 1844 | M. GOLDEN | BLDG 192, RM 121 |
| 1856 | D. BENIGNI | BLDG 17, RM 230 |
| 1863 | M. LAMATRICE | BLDG 17, RM 216 |
| 1864 | N. FICKEN | T-25 |
| 189 | G. GRAY | BLDG 17 |
| 1892.1 | J. STRICKLAND | BLDG 17, RM 100 |
| 1892.2 | D. SOMMER | BLDG 100, RM 1-F, ANNAPOLIS |
| 1892.3 | L. MINOR | BLDG 17, RM 105B |
| 1926 | D. VENDITIS | BLDG 15, RM 112 |
| 5641 | DTNSRDC LIBRARY | BLDG 1, RM 211 |
| 634 | B. PIERCE | BLDG 121, RM 132 |
| UEPD | V. POYNER | PORTSMOUTH, VA. |
| 6045F | J. GIVEN | NAVSEC - BLDG NC2 |
| 5033 | R. SAENGER | NAVAIR - BLDG JP2 |
| 4013 | S. BERNSTEIN | NAVAIR - BLDG JP2 |
| 5707 | H. BRYANT | NRL, BLDG 35 |
| | G. VEGA | BAY ST. LOUIS, MISS. |
| | W. SCHMIDT | CERL, CHAMPAIGN, ILL. |
| | S. STEFONCIK | NWSC, CRANE, IND. |

*** COMPUTER REFERENCE MANUALS ***

| NAME | PUBLICATION NUMBER |
|--|--------------------|
| COMPUTER CENTER POLICY | JULY 1975 |
| COMPUTER CENTER INTRODUCTORY REFERENCE MANUAL | CMLD-77-10 |
| COMPUTER CENTER REFERENCE MANUAL | CMLD-77-11 |
| COMPUTER CENTER LIBRARIES | CMLD-77-12 |
| COMPUTER CENTER LIBRARIES/NSRDC (SUBPROGRAMS) | CMLD-77-15 |
| COMPUTER CENTER LIBRARIES/PROFIL (PROCEDURES) | CMLD-77-16 |
| COMPUTER CENTER LIBRARIES/UTILITY (PROGRAMS) | CMLD-77-17 |
| COMPUTER OUTPUT MICROFICHE (COM) REFERENCE MANUAL | CMD-16-76 |
| ADP GLOSSARY | NAVSO P-3097 |
| COMPUTER PROGRAM DOCUMENTATION STANDARDS | SECNAV INST 5233.1 |
| EISPACK | |
| INTERNATIONAL MATHEMATICAL AND STATISTICAL LIBRARIES | |
| ABSTRACTS (IMSL) | |
| MATH SCIENCE LIBRARY ABSTRACTS (MSL) | |
| SC4020 PROGRAMMERS REFERENCE MANUAL | |
| VIM (SUPPLEMENT) (ON MICROFICHE) | |
| PROGRAMMING CALCOMP PEN PLOTTERS | 1006A 10M 1170 |
| CALCOMP FUNCTIONAL PACKAGE BUSINESS | 1011A 6C 1069 |
| CALCOMP FUNCTIONAL PACKAGE CRVT | 1014 6C 0769 |
| CALCOMP FUNCTIONAL PACKAGE DRAFTING | 1012A 3C 0670 |
| CALCOMP FUNCTIONAL PACKAGE GENERAL | 1013A 3C 0670 |
| CALCOMP FUNCTIONAL PACKAGE SCIENTIFIC | 1015B 6C 170 |
| CALCOMP THREE-D MANUAL | 1002B 5M 769 |
| DISSPLA POCKET MANUAL | |
| DISSPLA BEGINNER/INTERMEDIATE MANUAL VOLUME I | |
| DISSPLA ADVANCED MANUAL VOLUME II | |
| DISSFOP (DISSPLA POST-PROCESSOR OPTION) | |

| CDC MANUALS | PUBLICATION NUMBER |
|--|--------------------|
| AUTOMATIC PROGRAMMED TOOLING SYSTEM (APT IV) | |
| VERSION 2 REFERENCE MANUAL | 17326900 |
| BASIC VERSION 3 REFERENCE MANUAL | 19983900 C |
| COBOL VERSION 4 REFERENCE MANUAL | 60384100 F |
| | OR 60496800 B |
| COMPASS VERSION 3 REFERENCE MANUAL | 60360900 F |
| CYBER COMMON UTILITIES REFERENCE MANUAL | 60493300 B |
| | OR 60495600 D |
| FORM REFERENCE MANUAL | 60307000 F |
| FORTRAN EXTENDED VERSION 4 REFERENCE MANUAL | 60305600 K |
| | OR 60497800 B |
| FTN EXTENDED DEBUG USER'S GUIDE | 60329400 B |
| INTERACTIVE GRAPHICS SYSTEM | 17303600 F |
| INTERCOM VERSION 4 REFERENCE MANUAL | 60307100 F |
| | OR 60494600 C |
| INTERCOM INTERACTIVE GUIDE/COBOL USERS | 60385700 B |
| INTERCOM INTERACTIVE GUIDE/FTN USERS | 60359700 B |
| LOADER VERSION 1 REFERENCE MANUAL | 60344200 G |
| | OR 60429800 C |
| MARS VI VER. 2.1 (FULL INVERSION) | 17313000 A |
| MARS VI VER. 2.1 (PARTIAL INVERSION) | 17313100 A |
| MIMIC SIMULATION LANGUAGE | 44610400 D |
| NOS/RE 1 REFERENCE MANUAL | 60493800 C |
| PERT/COST REFERENCE | 17309200 |
| PERT/TIME REFERENCE | 60133600 B |
| RECORD MANAGER VERSION 1 REFERENCE (COMPASS) | 60307300 J |
| RECORD MANAGER GUIDE/COBOL USERS | 60385300 A |
| RECORD MANAGER VERSION 1 GUIDE/FTN USERS | 60385200 B |
| RECORD MANAGER USER'S GUIDE | 60359600 C |
| REFERENCE MANUAL | 60100000 AH |
| SCOPE VERSION 3.3/3.4 CONVERSION AIDS | 60358200 C |
| SCOPE 3.4 REFERENCE MANUAL | 60307200 L |
| SCOPE 3.4 USER'S GUIDE | 60358700 A |
| SIMSRIPT REFERENCE MANUAL | 60178300 C |
| SORT/MERGE VERSIONS 4 (AND 1) REFERENCE MANUAL | 60343900 J |
| | OR 60497500 B |
| UPDATE REFERENCE MANUAL | 60342500 G |

*** CENTRAL SITE EAM PROCESSING ***

AT THE CENTRAL SITE, CARD LISTING AND REPRODUCTION ARE PERFORMED WITHOUT ADDITIONAL CHARGE EXCEPT AS INDICATED BELOW.

- 1) EAM JOBS OF LESS THAN 2000 CARDS MAY BE PROCESSED DURING THE PRIME SHIFT.
- 2) EAM JOBS OF GREATER THAN 1999 CARDS WILL BE PROCESSED OVERNIGHT. JOBS OF THIS SIZE WHICH MUST BE PROCESSED DURING THE PRIME SHIFT WILL BE DONE WITH THE USER'S JOB ORDER NUMBER.
- 3) THE COMPUTER CENTER RESERVES THE RIGHT TO CHARGE AT THE ESTABLISHED RATE JOBS PROCESSING MORE THAN 4000 CARDS (LISTED OR REPRODUCED).
- 4) SYSTEMS UTILITIES WHICH SERVICE EAM FUNCTIONS MAY, OF COURSE, BE UTILIZED BY USERS. SEE USER SERVICES GROUP.

*** ADP CONTROL CENTER ***

THE ADP CONTROL CENTER HAS THE FOLLOWING CAPABILITIES:

- 1) REPRODUCE BINARY, BCD (026) OR EBCDIC (029) DECKS. THE DECKS CANNOT CONTAIN EMBEDDED END-OF-FILE (6/7/8/9) CARDS.
- 2) LIST DECKS. THE DECKS CANNOT CONTAIN EMBEDDED END-OF-FILE CARDS
- 3) CONVERT 029 DECKS TO 026. THE CONTROL CENTER MUST BE INFORMED IF THE INPUT IS EBCDIC. THE DECKS CANNOT CONTAIN EMBEDDED END-OF-FILE CARDS.

TO CONVERT 026 DECKS TO 029 DECKS, A SPECIAL PROCEDURE, CV29 (SEE CCLIP/P), IS AVAILABLE. THE DECKS TO BE CONVERTED CANNOT CONTAIN EMBEDDED END-OF-RECORD OR END-OF-FILE CARDS.

TO REPRODUCE A DECK WITH SEQUENCING, GANGPUNCHING OR ALTERATION OF FIELDS. (SEE CCLIP/U: CARDS/CARDS2)

THE FOLLOWING EAM FACILITIES ARE AVAILABLE OFF-LINE:

- 1) CARD INTERPRETER
- 2) DECOLLATOR (AVAILABLE FOR USE BY INDIVIDUALS)
- 3) SHREDDER (AVAILABLE FOR USE BY INDIVIDUALS)
- 4) CARD SORTER (AVAILABLE IN ANNAPOLIS)

*** PUNCHED CARDS FROM CENTRAL SITE ***

ALL END-OF-RECORD AND END-OF-FILE CARDS PUNCHED SHOULD HAVE BEEN MARKED BY ADP CONTROL WITH A RED MARKER ON THE END. IN ADDITION, ANY CARD WHICH TRIGGERED A COMPARE ERROR WILL BE MARKED. THE USER SHOULD PULL OUT ANY RED MARKED DEFECTIVE CARDS, SINCE THE CARD WAS REPUNCHED CORRECTLY BY THE SYSTEM IMMEDIATELY FOLLOWING THE BAD ONE.

***** USER HELPS *****

*** JOB RERUN ***

IF MACHINE FAILURE, END OF SHIFT, OR NEED FOR UNAVAILABLE RESOURCES MAKES RERUN OF A JOB NECESSARY, THE OPERATOR OR SYSTEM DEADSTART MAY INITIATE RERUN.

ANY JOB WHICH HAS PROPERLY CATALOGED, PURGED, EXTENDED, OR RENAMED A FILE MAY NOT BE RERUN BY OPERATOR TYPE-IN, BUT A SYSTEM DEADSTART MIGHT CAUSE ANY JOB TO BE RERUN.

WHENEVER POSSIBLE JOBS SHOULD BE SET UP SO THEY MAY BE RERUN. IF IT IS DESIRED TO INHIBIT THE POSSIBILITY OF RERUN WHICH MAY HAVE MANIPULATED TAPES TO SOME POSITION WHERE A RERUN MIGHT DESTROY THE CONTENTS, A SMALL FILE CAN BE CATALOGED AND THEN PURGED. THIS CAN BE ACCOMPLISHED BY 'BEGIN,NORERUN.'.

*** ABORT PROCESSING ***

CP TIME LIMIT

WHEN A JOB ABORTS ON "CP TIME LIMIT", FIVE CP SECONDS ARE ALLOCATED FOR "EXIT" PROCESSING. THIS ALLOWS TIME TO CATALOG FILES, UNLOAD, ETC. ONLY ONE SUCH TIME EXTENSION IS ALLOWED.

IO TIME LIMIT

WHEN A JOB ABORTS ON "IO TIME LIMIT", 100 IO SECONDS ARE ALLOCATED FOR "EXIT" PROCESSING. NO IO LIMIT IS ENFORCED UNLESS SPECIFIED BY USER.

MASS STORAGE LIMIT

WHEN A JOB ABORTS ON "MASS STORAGE LIMIT EXCEEDED", ENOUGH MASS STORAGE IS AVAILABLE FOR TAKING A DMP, AND ADDITIONALLY UP TO 200 PRUS ARE AVAILABLE FOR EXIT PROCESSING.

MESSAGE LIMIT EXCEEDED

THE DAYFILE CAN HAVE A MAXIMUM OF 1000 MESSAGES, AFTER WHICH THE JOB ABORTS.

REPRIEVE

WHEN REPRIEVE IS USED WITHIN A JOB TO CONTROL ABORT CONDITIONS, THE MAXIMUM ADDITIONAL ALLOCATION IS 5 CP SECONDS, 100 IO SECONDS, AND 200 PRUS MASS STORAGE. ONLY ONE SUCH REPRIEVE IS ALLOWED.

JUNE 1977

PAGE 14-2

*** CONVERSION AIDS ***

NON-STANDARD FORTRAN USAGE

THE ANSI FLAG OF THE FTN COMPILER ON THE CDC 6000 WILL INDICATE NON-STANDARD FORTRAN USAGE. USE FTN,EL=A,....

CONVERT DOUBLE PRECISION TO SINGLE PRECISION

CVT360 IS A SNOBOL PROGRAM TO CONVERT DOUBLE PRECISION S/360 FORTRAN PROGRAMS TO SINGLE PRECISION CDC FORTRAN. SEE CCRM, 10-30.

EBCDIC TO BCD

PUT '29' INTO COLUMNS 79-80 OF THE END-OF-RECORD (7/8/9) CARD PRECEDING THE DECK. THEN USE COPYCF,INPUT,PUNCH.

BCD TO EBCDIC

CV29 IS A PROCEDURE TO CONVERT FROM BCD TO EBCDIC. SEE CCLIB/P.

COBOL 4 TO COBOL 5

SEE CCRM, 5-16, FOR A DISCUSSION OF THE COBOL 4 TO COBOL 5 LANGUAGE CONVERSION SYSTEM (LCS).

SEE CONV MANUAL FOR ADDITIONAL AIDS.

*** COMMUNICATIONS WITH OPERATOR ***

*** "PM" CARRIAGE CONTROL DISPOSITION AT REMOTE SITES ***

"PM" IN COLUMNS 1 AND 2 OF AN OUTPUT PRINT LINE WILL STOP THE LINE PRINTER AND DISPLAY A MESSAGE AT THE OPERATOR'S CONSOLE OF 200UT TERMINALS (OR THE CENTRAL SITE) OR THE TELETYPE AT A 1700. THE MESSAGE (COLUMNS 3-32 OF THE PM PRINT LINE) SHOULD INFORM THE OPERATOR WHICH PRINT FORMS ARE TO BE MOUNTED OR CARRIAGE CONTROL TAPE IS REQUIRED BEFORE THE OPERATOR RESTARTS THE PRINTER FROM HIS CONSOLE. (SEE CCRM, 2-9 FOR TYPES OF SPECIAL FORMS.) SPECIAL FORMS USAGE AT CENTRAL SITE SHOULD ALWAYS BE BY ROUTE OR DISPOSE CARD (SEE CCRM, 2-10, 2-8).

REMOTE SITE SPECIAL FORMS MAY BE ACCOMPLISHED IN ONE OF TWO WAYS:

1. BY CONTROL CARD (SEE NOTE BELOW):

```

...
...
COPYCR(INPUT,OUTPUT)
...
(EXECUTE PROGRAM WHICH CREATES THE SPECIAL FORM OUTPUT)
...
EXIT,U.
COPYCR(INPUT,OUTPUT)
' 7/8/9 EOR
PM MOUNT SPECIAL FORMS TYPE
' 7/8/9 EOR
...
(RECORDS AS NEEDED FOR INTERMEDIATE STEPS)
...
' 7/8/9 EOR
PM REMOUNT STANDARD FORMS
" 6/7/8/9 EOF

```

2. WITHIN PROGRAM EXECUTION (SEE NOTE BELOW):

FORTRAN EXAMPLE

```

...
PRINT 10
10 FORMAT ("PM MOUNT SPECIAL FORMS TYPE")
...

```

NOTE: IT IS THE USER'S RESPONSIBILITY TO ENSURE THAT THE STANDARD FORMS OR CARRIAGE CONTROL TAPES ARE RESTORED TO THE PRINTER AT THE END OF THE JOB.

*** SPECIAL CARRIAGE CONTROL ***

WHEN THE FOLLOWING CHARACTERS ARE USED FOR CARRIAGE CONTROL (COLUMN 1) ON CENTRAL SITE OR 1700 PRINTERS, NO PRINTING TAKES PLACE. THE REMAINDER OF THE LINE IS IGNORED.

| CHARACTER | ACTION |
|-----------|--|
| Q | CLEAR AUTO PAGE EJECT (DEFAULT) |
| R | SELECT AUTO PAGE EJECT (SPACE OVER PAPER FOLD) |
| S | SET 6 LINES PER VERTICAL INCH (DEFAULT) |
| T | SET 8 LINES PER VERTICAL INCH. |

SEE CCRM, 14-3 FOR PM AND CCRM, 4-8, 5-5 FOR SPECIAL SPACING.

*** DAYFILE MESSAGE DISPLAY ***

TO DISPLAY A MESSAGE AT A CONTROL POINT AND IN THE DAYFILE:

| | |
|--------------|------------------------------|
| COROL | - DISPLAY |
| FTN | - CALL REMARK OR CALL DISPLA |
| CONTROL CARD | - COMMENT. |

THESE MESSAGES WILL NOT FLASH OR WAIT FOR OPERATOR ACTION.

*** RESERVED WORDS ***

FILES WHICH HAVE SPECIAL DISPOSITIONS AT END-OF-JOB MAY NOT BE USED AS FILE NAMES ON DEVICE SETS OR AS OVERLAY FILE NAMES.

| FILE NAME | MAY ONLY BE USED AS |
|-----------|----------------------------------|
| ----- | ----- |
| INPUT | JOB STREAM OR TTY INPUT |
| OUTPUT | JOB STREAM OR TTY OUTPUT |
| PUNCH | BCD PUNCH FILE |
| PUNCHB | BINARY PUNCH FILE |
| P80C | SPECIAL BINARY PUNCH FILE |
| FILMPL | SC4020 SCORS PLOT FILE |
| FILMPR | SC4060 META PLOT FILE |
| PLOT | 1700 REMOTE CALCOMP PLOT FILE |
| PTAPE | ASCII PAPER TAPE OUTPUT FROM APT |

INTERACTIVE INTERCOM WILL NOT ALLOW MOST SINGLE LETTER NAMES FOR OBJECT FILES SINCE THESE ARE ABBREVIATED INTERCOM COMMANDS.

INTERCOM USERS SHOULD AVOID ANY LOCAL FILES OF THE SAME NAME AS INTERCOM COMMANDS, SINCE MANY OF THESE HAVE A PRESET FIELD LENGTH WHICH COULD CAUSE A BINARY PROGRAM NOT TO LOAD.

***** INDEX *****

NOTE - NOS/BE SYSTEM CONTROL CARDS ARE FLAGGED WITH *.
 UPDATE DIRECTIVES BEGIN WITH *.

| A | PAGE |
|---------------------------------------|-----------------------|
| ABBREVIATION | I |
| ABNORMAL ENDING | 2-6 |
| ABORT | 4-1 , 4-8 , 14-1 |
| ABSOLUTE FILE | 5-1 |
| ABSOLUTE MODULE | 4-25 |
| ABSOLUTE OVERLAY | 7-15 |
| ACCESS LEVEL | 7-10 |
| ACCCUNT (AC) | 3-4 , 3-6 |
| ACCOUNTING | 2-4 |
| ADD, EDITLIB | 7-10 |
| *ADDFILE | 7-3 |
| ADP CONTROL CENTER | 13-1 , 13-5 |
| ADVANCED SCIENTIFIC COMPUTER | 12-15 |
| ADVANCING, COBOL | 5-8 |
| AEROSPACE RESEARCH LABORATORIES (ARL) | 10-3 |
| AETSKC | 8-2 |
| AEXEC | 8-2 |
| ALGOL | 11-1 |
| ALGORITHM | 11-1 |
| ALPHAMERIC | 3-6 |
| ALTEK AC-90 DIGITIZING SYSTEM | 12-14 |
| ALTER | 3-4 , 3-5 |
| ANALCG | 11-12, 12-14 |
| ANALYSIS OF VARIANCE | 10-22, 11-18 |
| ANALYSIS, STRUCTURAL | 11-11 |
| ANSI | 4-1 , 4-6 , 4-27, 5-2 |
| ANSI COBOL | 5-1 |
| ANSI, FORTRAN | 14-2 |
| APT3, APT4 | 11-2 |
| ARGCKNE | 10-3 , 10-22 |
| ARGUMENTS | 4-5 |
| ARLNALG | 10-3 |
| ARPANET | 12-16 |
| ARRAYS | 4-7 |
| ARRIBA | 10-22 |
| ASC | 12-15 |
| ASCII TERMINALS | 1-1 |
| ASCII 63-CHARACTER SET | 1-7 |
| ASCII6 | 1-7 , 5-8 |
| ASSEMBLER | 11-6 |

| A | (CONT) | PAGE |
|--------------------------------------|--------|-------------------------------|
| ASSETS, INTERCOM | | 9-4 |
| ATTACH | | 10-24 |
| "ATTACH | | 3-5 , 3-11, 4-16 |
| AUDIT | | 7-21 |
| "AUDIT | | 3-4 , 3-12 |
| AUDIT | | 7-22 |
| AUTO, TTY | | 9-5 |
| AUTCEDIT | | 8-4 |
| AUTOMATED DRAFTING | | 12-14 |
| AUTOMATIC PURGE | | 3-3 |
| AUTOMATICALLY PROGRAMMED TOOLS (APT) | | 11-2 |
| AXIS | | 10-7 |
| B | | PAGE |
| BACKSPACE | | 6-3 |
| BACKUP | | 9-9 |
| BANDIT | | 11-11 |
| BANNER | | 2-5 |
| "BASIC | | 9-6 , 10-23 |
| BASLIB | | 10-1 |
| BATCH | | 2-7 |
| BATCH JOB | | 1-3 |
| BATCH, INTERCOM | | 9-8 , 9-9 |
| BATCH, REMOTE | | 9-4 , 12-1 , 12-3 |
| BATCH, REMOTE COMMANDS | | 12-4 |
| BATCH, TELETYPE | | 2-8 |
| "BBTC6RM | | 4-11 |
| BCD | | 1-1 , 1-7 , 13-5 , 14-2 |
| "BEGIN | | 3-4 , 7-18, 7-21, 10-24, 11-3 |
| BEGIN, SEND | | 9-13 |
| BENDIX | | 11-2 , 12-14 |
| BESSEL FUNCTIONS | | 10-3 , 10-5 |
| BFS, BUFFER LENGTH | | 3-18 |
| BIMED | | 10-19, 10-24 |
| BIMEDP | | 10-19 |
| BINARY BLOCKED FILES | | 4-11 |
| BINARY FILE | | 6-1 |
| BINS, OUTPUT | | 2-5 |
| "BKSP | | 6-3 |
| BKY | | 9-21 |
| BLANK COMMON | | 4-10 |
| BLOCK ADDRESSES | | 12-11 |
| BLOCK CONTAINS | | 5-3 , 5-4 , 5-7 |
| BLOCK DATA | | 2-20, 4-9 |
| BLOCKING | | 3-13 |
| BRACKET | | 11-1 |
| BT, BLOCK TYPE | | 3-18 |

B (CONT)

PAGE

| | |
|-----------------------|------------------------------|
| BUFFER | 3-2 , 3-13, 4-3 , 4-8 , 4-11 |
| BUFFER | 4-14, 4-15, 10-8 , 10-13 |
| BUFFER IN | 4-10 |
| BUFFER IN, BUFFER OUT | 4-20 |
| BUILT-IN FUNCTIONS | 4-7 |
| | |
| BULLETIN, SYSTEM | 2-7 , 9-9 , 13-1 |
| BURSTABLE LIST | 4-1 |
| BURSTER | 13-5 |
| BYE, EDITOR | 9-9 |

C

PAGE

| | |
|------------------------|---------------------------------|
| CALCFN | 10-9 |
| CALCCMP | 8-9 , 10-11, 10-12, 10-17, 13-3 |
| CALCCMP PLOTTERS | 10-7 |
| CALCCMP 1700 | 2-8 , 10-8 , 12-1 |
| CALCCMP 763 | 12-11 |
| | |
| CALCCMP 936 | 12-10 |
| CALCULATOR | 11-20 |
| CALC170 | 10-8 |
| CALC3D | 10-12 |
| CALC763, CALC936 | 10-8 |
| | |
| *CALL | 7-3 |
| CALLS | 2-14 |
| CANCEL | 9-3 |
| CARD IMAGE | 3-13 |
| CARDS, CARDS2 | 13-5 |
| | |
| CARRIAGE CONTROL | 4-8 , 5-5 , 12-3 , 14-3 , 14-4 |
| CARRIAGE RETURN | 9-3 |
| CARRIAGE RETURN (CR) | 9-5 |
| "CATALOG | 3-5 , 3-11, 4-16, 9-14 |
| CATALOGUED PROCEDURES | 7-18, 7-20 |
| | |
| CATEGORIES OF SERVICE | 1-4 |
| CCRM | 10-20 |
| CCRM | 7-22 |
| CDC MANUALS | 13-4 |
| CDC 734 | 12-3 |
| | |
| CENTER NOTES | 13-1 |
| CENTRAL MEMORY (CM) | 1-1 , 2-3 |
| CENTRAL SITE | 2-8 , 13-5 |
| CENTREX | 9-1 |
| CERL | 2-21, 9-21, 11-12, 11-18 |
| | |
| CERTIFY | 3-14 |
| CHANGE FIELD LENGTH | 9-4 , 9-15 |
| CHANNELS | 1-1 , 3-13 |
| CHARACTER MANIPULATION | 10-5 |
| CHARACTER SET | 1-9 |

| C | (CONT) | PAGE |
|------------------------------|--------|---------------------------------|
| CHARACTRON | | 10-13 |
| CHARGE | | 3-3 , 13-5 |
| **CHARGE | | 2-1 , 2-4 |
| CHARGES | | 1-4 |
| CHECKPOINT | | 11-3 , 11-5 |
| CHEKPTX | | 11-3 |
| CIRCL | | 10-8 |
| CIRCUIT ANALYSIS | | 10-22 |
| CIVCO | | 10-22 |
| **CKP | | 11-3 |
| CLASSIFIED | | 1-4 |
| CLEANING, TAPE | | 3-14 |
| **CLEAR | | 2-6 |
| CLOSE FILES | | 4-16 |
| CLOSMS | | 4-8 |
| CM USED | | 2-13 |
| CM, CONVERSION MODE | | 3-18 |
| COBCL | | 3-19, 5-1 , 10-1 , 11-14, 11-19 |
| COBCL CONVERSION | | 14-2 |
| COBCL GENERATOR | | 11-17 |
| COBCL5 | | 5-14 |
| COBCL6 | | 1-7 |
| CODE CHECK | | 2-4 |
| CODED FILE | | 6-1 |
| COLLATING SEQUENCE | | 5-2 , 5-8 , 5-16 |
| COLOR, CARDS | | 2-1 |
| COLTMC | | 11-2 |
| COM | | 12-9 |
| **COMBINE | | 6-3 |
| *COMDECK | | 7-3 |
| COMMAND TIME LIMIT | | 9-15 |
| **COMMENT | | 2-6 , 14-4 |
| COMMENT, FORTRAN | | 4-9 |
| COMMON | | 2-19, 4-10 |
| COMMUNICATIONS WITH OPERATOR | | 14-3 |
| **COMPARE | | 6-3 |
| COMPARE ERROR | | 13-5 |
| COMPASS | | 4-9 , 11-6 |
| COMPATIBILITY | | 5-7 |
| *COMFILE | | 7-3 |
| COMPLAINTS | | 13-1 |
| COMPLEMENT | | 1-8 |
| COMPRESSED | | 7-2 |
| COMPRS | | 10-18 |
| COMPUTATIONAL | | 5-5 |
| COMPUTED GO TO | | 4-7 |
| COMPUTER CENTER NOTES | | 13-1 |

| C (CONT) | PAGE |
|-----------------------------------|-----------------------|
| COMPUTER USERS FORUM | 13-1 |
| COMPUTER-AIDED DESIGN ENVIRONMENT | 11-6 |
| COMRADE | 11-6 |
| CONDENSE | 7-15 |
| CONFIDENTIAL | 1-5 |
| CONNEC | 9-7 |
| CONNECT, INTERCOM | 9-4 , 9-7 |
| CONSOLE | 9-12 |
| CONSOLE SESSION, IGS | 8-6 |
| CONSULTATION | 13-1 |
| CONTENT, EDITLIB | 7-11 |
| CONTENTS, BINARY | 6-2 , 7-10 |
| CONTINUED CARD | 3-3 |
| CONTINUED DIRECTIVES | 2-20 |
| CONTINUOUS SYSTEM SIMULATOR | 11-12 |
| CONTICURING | 8-9 |
| CONTROL (CN) | 3-6 |
| CONTROL CARD | 1-13, 2-2 |
| CONTROL CARD PARAMETERS, COBOL | 5-1 |
| CONTROL CARD PARAMETERS, FTN | 4-1 |
| CONTROL CARDS | 1-4 , 7-18, 7-19, 9-8 |
| CONTROL CENTERS | 13-1 |
| CONTROL X | 9-3 |
| CONVERSION | 1-6 , 4-4 , 5-3 |
| CONVERSION AIDS | 14-2 |
| CONVERSION, NUMBERS | 1-11 |
| CONVERT | 6-9 , 10-21, 13-5 |
| COPY | 10-21, 10-21 |
| *COPY | 7-3 |
| **COPY | 6-3 |
| **COPYBF | 6-1 |
| COPYBFR | 6-1 , 10-21 |
| COPYBLK | 3-1 , 3-13 |
| **COPYBR | 6-1 |
| **COPYCF | 6-1 |
| **COPYCR | 6-1 |
| **COPYE | 6-4 |
| **COPYF | 6-5 |
| COPYFUL | 8-5 |
| **COPYL | 4-9 , 6-2 |
| **COPYLM | 6-2 |
| **COPYN | 4-9 , 6-3 |
| **COPYR | 6-5 |
| COPYRM | 3-1 , 6-9 |
| **COPYRM | 6-1 , 6-6 |
| COPYS | 6-1 , 10-21 |
| **COPYSBF | 6-6 |

| C | (CONT) | PAGE |
|-----------------|--------|------------------|
| "COPYSF, COPYSR | | 6-6 |
| COPY274 | | 8-5 |
| CORE | | 1-13, 2-3, 9-4 |
| CORE IMAGE | | 2-13 |
| CORE MAP | | 4-15 |
| COST | | 9-9 |
| CP TIME LIMIT | | 14-1 |
| CPA, CPB | | 2-3 |
| CPB | | 4-16 |
| "CPINDEX | | 7-20 |
| CPU | | 1-3, 2-3 |
| CPU TIME | | 9-4 |
| CREATE, EDITOR | | 9-6 |
| CREATION | | 7-2 |
| CROSS REFERENCE | | 4-13, 4-15 |
| CTRL H | | 9-3 |
| CTRL S | | 9-3 |
| CURVE FIT | | 10-5 |
| CVT360 | | 4-4, 10-21, 14-2 |
| CV29 | | 1-6, 13-5, 14-2 |
| CYCLE (CY) | | 3-6 |

| D | PAGE |
|------------------------------------|---------------------------------|
| DATA BASE | 11-7, 11-8, 11-12, 11-17, 11-20 |
| DATA BASE MANAGEMENT SYSTEM (DBMS) | 11-13, 11-19 |
| DATA MANAGEMENT | 11-6, 11-7 |
| DATA REDUCTION | 12-14 |
| DATA SETS | 9-1 |
| DATA 100 | 12-3 |
| DATAGRAPHIX 4530 | 12-9 |
| DAY,OFF | 9-15 |
| DAYFILE | 1-4, 2-5, 2-21, 4-6, 4-9 |
| DAYFILE | 4-14, 4-15, 5-2, 9-13, 14-1 |
| DAYFILE | 14-4 |
| DB, COBOL | 5-9 |
| DC, ROUTE | 2-10 |
| DDL, DATA DESCRIPTION LANGUAGE | 11-7 |
| DEBUG | 4-3, 4-15 |
| DEBUG (D) | 4-1 |
| DEBUG, COBOL | 5-6 |
| DECK | 9-9, 13-5 |
| *DECK | 7-3 |
| DECODE | 4-8, 4-20 |
| DECOLLATOR | 13-5 |
| DEFAULT LENGTH | 4-3 |
| DEFERRED ROUTING (DEF) | 2-11 |
| *DELETE | 7-3 |

| D | (CONT) | PAGE |
|------------------------|--------|-----------------------------|
| DELETE, EDITLIB | | 7-10 |
| DELETE, EDITOR | | 9-3 , 9-7 |
| DELIMIT | | 4-5 |
| DENSITY, TAPE | | 3-13, 3-16 |
| DEPENDENCY | | 2-3 , 2-7 |
| DEVICE SETS | | 2-3 , 3-1 , 3-10, 14-4 |
| DIAGNOSTICS | | 4-1 , 4-13, 4-14, 4-27, 5-2 |
| DIEROLD TAPE | | 3-14 |
| DIFFERENTIAL EQUATIONS | | 10-4 , 11-12 |
| DIGIGRAPHICS 274 | | 8-1 |
| DIGITAL | | 11-12 |
| DIGITIZER | | 8-9 , 12-14 |
| DIRECTIVES, EDITLIB | | 7-9 |
| DIRECTIVES, SEGLOAD | | 2-19 |
| DIRECTIVES, UPDATE | | 7-3 |
| DIRT, TAPE | | 3-14 |
| DISCARD | | 9-14 |
| DISCONNECT | | 9-3 |
| DISCONT, INTERCOM | | 9-7 |
| DISK PACK | | 3-2 , 3-10 |
| DISPLAY | | 14-4 |
| DISPLAY CODE | | 1-7 , 1-9 , 2-1 |
| DISPLAY, PAGE | | 9-19 |
| "DISPOSE | | 2-8 , 9-4 , 9-8 |
| DISPOSITION | | 2-10, 14-3 |
| DISSPLA | | 10-17, 13-3 |
| DIVERT | | 12-6 |
| DIVIDE CHECK | | 4-7 |
| DIVISION BY ZERO | | 4-4 , 4-5 |
| "DMP | | 2-6 |
| DMPX | | 2-21, 4-15 |
| DMS170 | | 11-7 |
| DO LOOPS | | 4-7 |
| DOCUMENTATION | | 4-16, 7-21, 10-24 |
| DOSSIER | | 2-7 , 4-16 |
| DOUBLE BUFFERING | | 4-3 |
| DOUBLE PRECISION | | 4-4 , 4-5 , 10-21 |
| DOUBLE-QUOTE | | 4-10 |
| DRAFTING, AUTOMATED | | 12-14 |
| DRAFTING, CALCOMP | | 10-9 |
| "DSMCUNT | | 3-11 |
| DUM | | 3-10 |
| DUMMY ARGUMENTS | | 4-5 |
| DUMP | | 2-6 , 2-21 |
| DUMP, FORTRAN | | 4-15 |
| "DUMPF | | 3-12 |
| "DUP | | 7-19 |

| D (CONT) | PAGE |
|-------------------------|------------------------------|
| DUPLICATE CONTROL CARDS | 7-19 |
| DYNAMIC | 4-22 |
| DYNAMIC LOAD | 2-14 |
| E | PAGE |
| E, COBOL | 5-1 |
| EAM | 13-5 |
| EBCDIC | 1-6 , 3-16, 3-17, 4-4 , 6-9 |
| ERCDIC | 13-5 , 14-2 |
| ECAP | 10-22 |
| EDIT | 8-4 |
| EDIT, EDITOR | 9-7 |
| EDITLIB | 2-15, 5-1 , 7-9 , 7-12, 7-15 |
| EDITOR | 9-3 , 9-5 , 9-6 , 9-9 |
| EDITOR, INTERCOM | 9-8 |
| EDITOR, TEXT | 9-21 |
| EDSTAT | 10-3 |
| EFL | 9-15 |
| EIGENSYSTEM | 10-5 |
| EIGENVALUES | 10-3 |
| EISPACK | 10-3 |
| ELLIPTIC INTEGRALS | 10-3 , 10-5 |
| ENCODE | 4-8 , 4-20 |
| END AROUND SEARCH | 6-3 |
| END-CF-FILE | 4-7 , 5-9 , 9-3 |
| END-CF-FILE (6/7/8/9) | 2-1 |
| END-CF-INFORMATION | 9-3 |
| END-CF-JOB | 2-7 |
| END-CF-RECORD | 9-3 |
| END-CF-RECORD (7/8/9) | 1-6 , 2-1 |
| ENCRUN, EDITLIB | 7-11 |
| ENDUP | 7-19 |
| ENTRY | 4-6 |
| *EOF | 9-3 |
| EOF | 4-7 |
| EOP, ALGOL | 11-1 |
| *EOR | 9-3 |
| ERR, TAPE | 2-21 |
| ERROR FILE | 3-19 |
| ERROR FUNCTION | 10-5 |
| ERROR MESSAGES, FORTRAN | 4-14 |
| ERROR MODE | 1-8 |
| ERRORS | 6-3 |
| ERRORS, COBOL | 5-2 |
| ERRORS, FORTRAN | 4-15 |
| ERRORS, LOADER | 2-21, 9-3 |
| ERRORS, MODE | 2-6 , 2-21, 4-15 |

| E (CONT) | PAGE |
|------------------------------|-------------------------------|
| ERRORS, NOS/BE | 4-15 |
| ERRORS, PERMANENT FILE | 3-7 |
| ERRSET | 4-14, 4-16 |
| ESC KEY | 9-3 |
| "ESTIMATE | 3-19 |
| ETL | 9-15 |
| EVEN PARITY | 6-1 |
| EXACT COPY | 6-5 |
| EXECUTE | 9-16 |
| "EXECUTE | 2-15 |
| EXECUTION ERROR | 5-9 |
| "EXIT | 2-6, 14-1 |
| "EXIT(S), EXIT(U) | 2-6, 7-18 |
| EXPIRATION DATE | 3-3 |
| EXPCNENT | 1-8 |
| EXTEND | 3-9, 4-12 |
| "EXTEND | 3-5 |
| EXTEND (EX) | 3-6 |
| EXTERNAL | 2-15, 4-5 |
| F | PAGE |
| FACTORIZATION | 10-3 |
| FALSE | 1-8 |
| FARRINGTON | 12-17 |
| FASTIN | 10-5 |
| FATAL | 2-21 |
| FATAL ERROR | 4-14 |
| FC, ROUTE | 2-11 |
| FEE | 3-3 |
| FETCH | 9-14 |
| FID, ROUTE | 2-11 |
| FIELD LENGTH | 1-8, 2-18, 9-4, 9-15 |
| FIELD LENGTH (FL) | 1-4, 1-13, 1-14 |
| FIELDATA | 5-16 |
| FILE | 3-1, 4-12, 11-8 |
| "FILE | 3-13, 3-18, 3-19, 4-11, 11-14 |
| FILE DESCRIPTION, COBOL | 5-4 |
| FILE IDENTIFICATION | 2-11 |
| FILE INFORMATION TABLE (FIT) | 3-18 |
| FILE MANIPULATION | 2-7, 2-15, 6-1, 7-10, 10-5 |
| FILE NAME SUBSTITUTION | 5-6 |
| FILE ORGANIZATION | 3-18 |
| FILE REPLACEMENT | 4-3 |
| FILE SIZE | 3-3 |
| FILES, INTERCOM | 9-4, 9-8 |
| FILES, SAVE | 9-14 |

F (CONT)

PAGE

| | |
|-------------------|-----------------------------|
| FILES, XEQ | 9-16 |
| FILM | 12-8 , 12-9 |
| FILMPL | 2-10, 10-13, 14-4 |
| FILMPR | 10-16, 14-4 |
| FINISH, EDITLIB | 7-10 |
| FINITE ELEMENT | 11-11 |
| FLAG | 2-18 |
| FLAW | 3-10 |
| FLOATING POINT | 1-8 , 1-11 |
| FMILL | 11-2 |
| FNPLT | 10-14 |
| "FORM | 3-1 , 3-19, 3-20, 4-16, 6-1 |
| FORMAT INVALID | 4-7 |
| FORMAT, EDITOR | 9-6 |
| FORMS | 2-9 |
| FORMS CODES | 2-11 |
| FORM1, FORM2 | 4-16 |
| FORTRAN | 1-7 , 2-15, 4-1 , 4-26, 9-6 |
| FORTRAN | 9-7 , 10-1 , 11-14, 11-19 |
| FORTRAN ERRORS | 4-15 |
| FORTRAN 77 | 4-27, 4-28 |
| FOURIER TRANSFORM | 10-5 |
| FRACTIONS | 1-12 |
| FRAMEV | 10-14 |
| "FTN | 4-1 |
| FTNBIN | 4-11 |
| FUNCTIONS | 4-7 |
| FUNDS | 2-4 |
| FUNPACK | 10-3 |

G

PAGE

| | |
|------------------------|-----------------|
| GAMMA FUNCTION | 10-5 |
| GECENT | 11-2 |
| GET | 4-11 |
| GIMAIN | 8-2 |
| GIMESG | 8-2 |
| GILOT | 8-5 |
| GIVING, COBOL | 5-5 |
| GLOBAL | 2-17, 2-19 |
| GO | 4-1 |
| GO TO-LESS | 4-28 |
| GPSS | 11-17 |
| GRAPHIC DATA REDUCTION | 12-14 |
| GRAPHIC INPUT (GID) | 8-9 |
| GRAPHICS | 12-1 , 12-8 |
| GRAPHICS, INTERACTIVE | 1-3 , 8-1 , 8-8 |

| G | (CONT) | PAGE |
|----------------------|--------|-------------------------|
| GRAPHING | | 10-16 |
| GRDS | | 12-14 |
| GRIPE | | 13-1 |
| GRIPE | | 7-22 |
| GSA | | 12-13 |
| H | | PAGE |
| HARDWARE | | 1-1 |
| HARDWIRED | | 1-1 |
| HELP, PAGE | | 9-18 |
| HEXADECIMAL | | 4-4 |
| HOLLERITH | | 1-7 , 4-5 |
| HOLLERITH STRINGS | | 4-10 |
| HOUSTON PLOTTER | | 10-23, 12-3 |
| I | | PAGE |
| IBM S/360 | | 5-3 |
| ID | | 3-4 |
| ID FRAME | | 10-13, 10-16 |
| IDEAL POST PROCESSOR | | 8-9 |
| *IDENT | | 7-3 |
| IDENTIFIER | | 7-2 |
| IF-THEN-ELSE | | 4-27 |
| IF, FTN | | 4-7 |
| IGS | | 8-1 , 8-6 , 12-1 |
| IGS274 | | 10-17 |
| IGS274 | | 10-1 |
| ILLEGAL INSTRUCTION | | 2-21 |
| IMAGE POST PROCESSOR | | 8-9 |
| IMPLICIT (FTN) | | 4-10 |
| IMSL | | 10-4 |
| INCLUDE | | 2-19, 2-20 |
| INDEFINITE | | 1-8 , 2-21, 4-4 |
| INDEXED SEQUENTIAL | | 3-1 , 3-19, 5-6 , 11-20 |
| INFINITE | | 1-8 , 2-21, 4-4 |
| INFINITE RETENTION | | 3-3 |
| INFORMATIVE | | 4-14 |
| INITIALIZE | | 3-10, 5-9 |
| INITIALS, REGISTERED | | 2-3 |
| INPUT | | 4-3 , 14-4 |
| INPUT FILE | | 5-4 |
| INPUT QUEUE | | 1-4 , 2-4 , 2-10, 2-11 |
| *INSERT | | 7-3 |
| INSTALLATION | | 2-3 |
| INSTRUCTION | | 1-8 |
| INTEGER | | 1-8 |

| I (CONT) | PAGE |
|-----------------------------|------------------------------|
| INTEGER MULTIPLICATION | 4-5 |
| INTEGER PROGRAMMING | 10-22 |
| INTEGRATION | 10-5 |
| INTERACTIVE | 1-3 , 8-8 , 9-1 |
| INTERACTIVE DATA MANAGER | 8-4 |
| INTERACTIVE GRAPHICS | 1-3 , 8-1 |
| INTERCOM | 1-3 , 2-8 , 3-4 , 3-7 , 4-3 |
| INTERCOM | 9-1 , 11-6 , 12-3 , 14-4 |
| INTERNAL BCD | 1-7 |
| INTERPOLATION | 10-4 |
| INTERPRETER | 13-5 |
| INVALID | 3-3 |
| INVALID NUMBERS | 4-7 |
| INVERSION | 11-8 |
| INVERSION, FILE | 11-13 |
| IO TIME LIMIT | 14-1 |
| IOCHECK | 4-7 |
| ISDS | 11-6 |
| "ITEMIZE | 6-2 |
| ITIS.2 | 8-5 |
| J | PAGE |
| JACOBIAN ELLIPTIC FUNCTIONS | 10-5 |
| JANUS | 1-3 |
| JOB | 1-3 , 1-6 , 2-3 , 9-10, 9-11 |
| JOB DESCRIPTOR (JDT) | 1-4 |
| JOB ORDER | 2-4 |
| JOB PROCESSING | 2-1 |
| JOB RERUN | 14-1 |
| K | PAGE |
| KEY | 11-9 |
| KEYPUNCH | 1-6 |
| KEYWORDS | 2-2 , 3-6 |
| L | PAGE |
| L-TAPE | 3-13, 3-17 |
| LABEL | 3-16 |
| "LABEL | 3-13, 10-24 |
| LABEL RECORDS, COBOL | 5-4 |
| LCS, COBOL 4 TO COBOL 5 | 14-2 |
| LDSET | 3-13, 10-2 |
| "LDSET | 1-8 , 2-16, 3-18, 4-11, 9-16 |
| "LDSET | 10-19 |
| LEFT DISPLAY CODE | 4-21 |
| LEGVAR | 4-4 , 4-7 |

| L | (CONT) | PAGE |
|-------------------------------|--------|--------------------------------|
| LENGTH | | 4-8 , 4-10, 4-20 |
| LEVEL | | 4-22 |
| LEVEL 15 | | 5-9 |
| LIB | | 2-16 |
| "LIBLOAD | | 2-15, 9-16 |
| LIBRARIAN, TAPE | | 1-4 , 3-16 |
| LIBRARIAN,TAPE | | 3-10 |
| LIBRARIES | | 7-1 |
| LIBRARIES OF MAIN PROGRAMS | | 10-19 |
| LIBRARIES OF SUBPROGRAMS | | 10-2 |
| LIBRARIES, SYSTEM | | 10-1 |
| LIBRARY | | 5-1 , 7-15, 9-4 , 10-1 , 12-13 |
| "LIBRARY | | 2-17, 4-11, 10-19 |
| LIBRARY OF BINARY ROUTINES | | 7-9 |
| LIBRARY, CONDENSE | | 7-13 |
| LIBRARY, EDITLIB | | 7-10 |
| LIGHT PEN | | 8-6 |
| "LIMIT | | 2-6 |
| LIMIT, TIME, STORAGE, MESSAGE | | 14-1 |
| LIMITATIONS, INTERCOM | | 9-4 |
| LINE | | 10-7 |
| LINE FEED | | 9-3 , 9-5 |
| LINE LENGTH, INTERCOM | | 9-15 |
| LINE LIMIT (PL) | | 2-15, 4-2 , 4-10 |
| LINE NUMBER | | 9-6 |
| LINE TERMINATOR | | 3-2 |
| LINEAR ALGEBRA | | 10-3 , 10-4 |
| LINEAR EQUATIONS | | 10-5 |
| LIST | | 13-5 |
| LIST DIRECTED I/O | | 4-10 |
| LIST-DIRECTED | | 4-3 |
| LIST, EDITOR | | 9-6 |
| LISTING OPTIONS (LO) | | 5-15 |
| LISTLIB, EDITLIB | | 7-11 |
| LITERALS | | 5-3 |
| LOAD | | 9-16 |
| "LOAD | | 2-13, 2-15 |
| LOAD MAP | | 2-17 |
| LOADER | | 2-13, 2-21, 4-15 |
| LOADER ERRORS | | 2-21, 9-3 |
| LOADING, AUTOMATIC | | 7-9 |
| "LOADPF | | 3-12 |
| LOCAL | | 2-17 |
| LOCK, INTERCOM | | 9-13 |
| LOCK,ON | | 9-12 |
| LOCKIN | | 11-5 |
| LOGARTITHMIC PLOT | | 10-9 |

| L | (CONT) | PAGE |
|-------------------------|--------|------------------------------|
| LOGICAL | | 1-8 |
| LOGICAL RECORD | | 1-4 , 1-6 |
| LOGIN | | 2-7 , 9-1 , 9-3 , 9-4 |
| LOGIN, INTERCOM | | 9-9 |
| LOGOUT INTERCOM | | 9-9 |
| LOST TIME | | 2-1 |
| LOWEST CYCLE (LC) | | 3-6 |
| M | | PAGE |
| M, INTERCOM MESSAGE | | 9-13 |
| MACHINE TOOLS | | 11-2 |
| MACSYMA | | 12-16 |
| MAIL | | 12-16 |
| MAIN PROGRAM | | 4-21 |
| MAINFRAME | | 1-1 |
| MANUAL | | 10-20 |
| MANUALS, REFERENCE | | 13-3 |
| MAP | | 2-16 |
| "MAP | | 2-17 |
| MAP, CROSS REFERENCE | | 4-2 , 4-15, 5-2 |
| MAP, LOAD | | 2-13, 2-17, 4-15 |
| MARS VI | | 9-4 , 11-8 |
| MASS STORAGE | | 3-1 |
| MASS STORAGE LIMIT | | 2-6 , 14-1 |
| MASTER FILE | | 6-1 |
| MASTER PACK | | 3-10 |
| MATH LIBRARY | | 4-14 |
| MATH SCIENCE LIBRARY | | 10-4 |
| MATRIX | | 10-4 |
| MEDIUM SPEED | | 9-1 , 12-3 |
| MERGE | | 6-3 , 7-2 , 11-14 |
| MESSAGE | | 9-12, 9-13, 14-4 |
| MESSAGE LIMIT | | 14-1 |
| META | | 10-16, 10-17, 12-8 |
| MICROFICHE | | 12-9 |
| MIMIC | | 11-12 |
| MINNESOTA FORTRAN (MNF) | | 4-26 |
| MIXED-MODE | | 4-5 |
| MNF | | 4-27 |
| "MODE | | 2-6 |
| MODE ERROR | | 2-6 , 2-21, 4-6 , 4-14, 4-15 |
| MODE, TAPE | | 3-13 |
| MODIFY (MO) | | 3-6 |
| MOHAWK | | 12-3 |
| "MOUNT | | 3-11 |
| MOVE ARRAYS (FTN) | | 4-10 |
| MOVLEV | | 4-10 |

| M (CONT) | PAGE |
|--------------------------------------|---------------------|
| MPS | 12-9 |
| MR, ATTACH | 14-1 |
| MRL, MAXIMUM RECORD LENGTH | 3-18 |
| MSL | 10-4 |
| MT/NT COUNT | 2-7 |
| MULTI-READ (MR) | 3-6 |
| MULTIPLE ENTRY | 4-6 , 4-9 |
| MULTIPLE RETURNS | 4-6 |
| MULTIPLEXORS | 1-1 |
| MULTIREEL FILE | 3-15 |
| N | PAGE |
| NALCON | 12-16 |
| NAMED PROCEDURES | 7-18, 7-20 |
| NAMELIST | 4-3 , 4-8 |
| NARROW PAPER | 2-9 |
| NASTRAN | 11-11 |
| NAVY LABORATORIES COMPUTER NETWORK | 12-16 |
| NETED | 9-21 |
| NETWORK | 1-3 , 12-16 |
| NEWPEN | 10-8 , 12-10 |
| NEWPL | 7-2 |
| NINE-TRACK (NT) | 3-13 |
| NO-OPERATION INSTRUCTION | 1-8 |
| "NOGO | 2-15, 9-16 |
| NOL | 10-23 |
| NON-NUMERIC, COBOL | 5-9 |
| NONLINEAR EQUATIONS | 10-4 |
| NORING, TAPE | 3-16 |
| NORTHWESTERN UNIVERSITY | 10-21, 11-12, 11-18 |
| NOS/EE | 5-7 |
| NOS/EE ERRORS | 4-15 |
| NOVA | 12-3 |
| NOVACOM | 10-22 |
| NSRDC | 4-16, 10-24 |
| NSRDC, ROUTINES ON | 10-5 |
| NUCLEUS | 2-17, 10-1 |
| NUMBER | 10-7 |
| NUMERICAL ANALYSIS | 11-17 |
| NUMERICALLY CONTROLLED MACHINE TOOLS | 11-2 |
| NWL | 10-22 |
| O | PAGE |
| OBJECT MODULES | 2-13, 4-12 |
| OCP | 12-17 |
| OCTAL | 1-7 , 4-5 |
| OCTAL-DECIMAL CONVERSION | 1-11 |
| ODD PARITY | 6-1 |

| O (CONT) | PAGE |
|-------------------------------|------------------------------|
| OFF-LINE REQUEST | 12-8 |
| OLDLIB | 6-2 |
| OLDPL | 7-2 , 10-21 |
| ON-LINE | 3-2 |
| ONMITAR | 11-17 |
| OPENMS | 4-8 |
| OPERATING SYSTEM | 1-3 |
| OPTICAL CHARACTER RECOGNITION | 12-17 |
| OPTIMIZATION (OPT) | 4-2 |
| OPTION, LOSET | 2-16 |
| OPTICNS | 9-16 |
| OUTPUT | 2-5 , 2-8 , 2-10, 4-3 , 14-4 |
| OUTPUT FILE | 4-10, 5-4 |
| OUTSIDE USERS | 13-1 |
| OVERFLOW | 4-7 |
| OVERLAY | 2-14, 2-16, 3-1 , 4-9 , 4-10 |
| OVERLAY | 4-21, 4-23, 4-25, 5-12, 8-2 |
| OVERLAY | 10-8 , 10-14 |
| OVERLAY FILE NAME | 7-15 |
| OVERLAY IN A LIBRARY | 7-15 |
| OVERLAY, ABSOLUTE | 7-15 |
| OVERLAY, COBOL | 5-12 |
| OVERLAY, FORTRAN | 4-21 |
| OVERRIDE, FILES | 2-15 |
| OVLNAME | 7-15 |
| P | PAGE |
| PACKS | 3-2 |
| PAGE, INTERCOM | 9-8 , 9-18 |
| PAGEPRT | 10-20 |
| PAPER TAPE | 2-8 , 11-2 , 12-1 |
| PAPER TAPE, INTERCOM | 9-5 |
| PAPER, CALCOMP | 12-12 |
| PARAMETER | 2-2 , 4-27 |
| PARAMETER SUBSTITUTION | 7-18 |
| PARITY, TAPE | 2-21, 3-13 |
| PASSWORD (PW) | 3-6 |
| PASSWORD, INTERCOM | 9-1 |
| PASSWORDS | 3-10 |
| "PAUSE | 3-11 |
| PCDC | 12-8 |
| PERCENT A | 9-3 |
| PERCENT EOF | 9-3 |
| PERCENT EOR | 9-3 |
| PERCENT S | 9-3 |
| PERIPHERAL PROCESSORS (PP) | 1-1 |
| PERMANENT FILES | 1-3 , 1-4 , 3-3 , 3-10, 9-14 |

| P | (CONT) | PAGE |
|----------------------------|--------|----------------------------------|
| PERSONNEL | | 11-6 |
| PERT, PERTC, PERT66 | | 11-12 |
| PERTC, PERT66 | | 1-14 |
| PHOSPHOR, SCREEN | | 8-8 |
| PHYSICAL RECORD UNIT (PRU) | | 3-2 , 3-13 |
| PL/I | | 11-17 |
| PL, FORTRAN | | 2-15, 4-2 , 4-10 |
| PLFILE | | 10-18 |
| PLOT | | 10-7 , 10-8 , 10-18, 12-11, 14-4 |
| PLOT 10 | | 8-8 |
| PLOTPR | | 10-7 |
| PLOTS, PRINTER | | 10-7 |
| PLOTTERS, CALCOMP | | 10-7 , 10-9 |
| PLOTTERS, OFF-LINE | | 10-7 |
| PLOTTING | | 10-5 , 10-13, 10-17, 12-10 |
| PLTND | | 10-13 |
| PL1 | | 11-17 |
| PM | | 2-10, 4-8 , 14-3 |
| POLAR PLOT | | 10-9 |
| POLICY | | 1-4 , 2-3 |
| POLYNOMIAL | | 10-4 , 10-9 |
| POLYNOMIAL ROOTS | | 10-5 |
| POP936, POP4060, POP274 | | 10-18 |
| PORT | | 9-1 |
| POSITION | | 6-1 |
| POSITIONAL PARAMETERS | | 2-2 |
| POST-PROCESSOR | | 10-18 |
| PRE-COMPILER | | 4-28 |
| PRECISION | | 1-7 |
| PRELOAD | | 8-4 |
| PRESFT | | 1-8 |
| PRESET, PRESETA | | 2-16 |
| PRESETA | | 9-16 |
| PRINT DENSITY (PD) | | 5-15 |
| PRINT FORMS | | 2-9 |
| PRINT LIMIT (PL) | | 2-15, 4-2 , 4-10 |
| PRINTER PLOT | | 11-20 |
| PRINTER PLOTS | | 10-7 |
| PRICR | | 12-6 |
| PRICPITY | | 1-4 , 2-3 , 2-4 |
| PRIVACY | | 1-5 |
| PRIVATE | | 1-5 , 2-5 |
| PRIVATE PACK | | 3-10 |
| "PRNTSPY | | 2-7 |
| PROCEDURE | | 5-5 , 11-1 |
| PROCEDURE DIVISION, COBOL | | 5-4 |
| PROCEDURE FILE | | 7-21 |

| P | (CONT) | PAGE |
|-----------------------|--------|----------------------------|
| PROCEDURES | | 7-18 |
| PROCEDURES, NAMED | | 7-20 |
| PROCEDURES, UNNAMED | | 7-20 |
| PRODUCTS | | 1-14 |
| PROFIL | | 10-24 |
| PROFIL | | 7-21 |
| PROGDOC | | 4-27 |
| PROGDOC | | 10-24 |
| PROGRAM CARD, FTN | | 2-15, 4-3, 4-8, 4-11 |
| PROGRAM CONSTRUCTION | | 9-6 |
| PROGRAM DOCUMENTATION | | 10-24 |
| PTAPE | | 14-4 |
| PUBLIC | | 3-6, 10-2 |
| PUBLIC LAW 93-579 | | 1-5 |
| *PULLMOD | | 7-3 |
| PUNCH | | 2-10, 4-3, 9-4, 13-5, 14-4 |
| PUNCH CARD | | 2-9 |
| PUNCHR | | 4-1, 4-18, 5-1, 14-4 |
| PUNTAP | | 11-2 |
| *PURCECK | | 7-3 |
| PURGE | | 3-3 |
| *PURGE | | 7-3 |
| **PURGE | | 3-5 |
| PUT | | 4-11 |
| P80C | | 14-4 |

| Q | PAGE |
|----------------------|--------------|
| Q, INTERCOM | 9-8, 9-10 |
| QS, INTERCOM | 9-11 |
| QUERY | 11-12, 11-13 |
| QUERY UPDATE | 11-7 |
| QUEUE | 1-4, 2-4 |
| QUEUE DEVICE | 2-8, 2-10 |
| QUEUE RESIDENT FILES | 12-6 |
| QUOTE | 11-1 |
| QUOTES | 4-5, 5-5 |
| QUOTES, DOUBLE | 5-3 |
| QWICK QWERY | 11-12 |

| R | PAGE |
|---------------------------|-----------------------------|
| R, TAPE LABEL | 3-16 |
| RANDCM | 6-1, 7-2, 7-11, 7-20, 10-21 |
| RANDCM ACCESS | 3-1, 3-2, 4-8 |
| RANDCM FILE | 11-9 |
| RANDCM NUMBERS | 10-4, 10-5 |
| RANTCSEQ, EDITLIB | 7-11 |
| RATIONAL FORTRAN (RATFOR) | 4-28 |

| R (CONT) | PAGE |
|------------------------------------|------------------------------|
| RB, RECORD BLOCKING | 3-18 |
| *READ | 7-3 |
| READ-ONLY (XR) | 3-6 |
| READMS | 4-8 , 4-12 |
| READY.. | 9-18 |
| RECORD BLOCKS (RB) | 3-2 |
| RECORD IDENTIFICATION CARDS, COPYN | 6-3 |
| RECORD LEVEL | 6-5 |
| RECORD MANAGER | 2-16, 2-20, 2-21, 3-19, 4-11 |
| RECORD MANAGER | 5-6 , 5-7 , 6-6 |
| RECORD MANAGER (RM) | 3-1 |
| *RECOVER | 3-12 |
| RECOVERABLE | 3-3 |
| RECOVER | 4-8 , 4-14, 4-16 |
| *REDUCE | 2-18 |
| REFERENCE CENTERS | 13-2 |
| REFERENCE MANUAL | 10-20 |
| REFERENCE MANUALS | 12-2 , 13-3 |
| REFORMATTING IN CORE | 4-8 |
| REFUND | 13-1 |
| REFUNDS | 2-1 |
| REGISTERS | 1-3 |
| REGRESSION ANALYSIS | 10-4 , 10-19, 11-18 |
| RELEASE | 2-8 |
| RELOCATABLE MODULE | 7-9 |
| REMARK | 14-4 |
| REMOTE BATCH | 9-1 , 9-4 , 12-1 , 12-3 |
| REMOTE BATCH COMMANDS | 12-4 |
| RENAME | 3-9 |
| *RENAME | 3-5 |
| REP, ROUTE | 2-11 |
| REPLACE, EDITLIB | 7-10 |
| REPORT GENERATION | 11-13, 11-17, 11-20 |
| REPRIEF | 14-1 |
| REPRIEF (ER) | 4-1 , 4-12, 4-14 |
| REPRODUCE | 13-5 |
| *REQUEST | 2-7 , 3-5 , 3-11 |
| REQUEST | 4-16 |
| *REQUEST, TAPE | 3-17 |
| REPUN | 14-1 |
| RESERVED WORDS | 5-7 , 14-4 |
| RESET | 8-8 |
| RESOURCES | 1-4 |
| *RESTART | 11-3 , 11-5 |
| *RESTORE | 7-3 |
| RETENTION (RP) | 3-6 , 3-10 |
| RETENTION, FILES | 3-3 |

| R | (CONT) | PAGE |
|-------------------------------------|--------|-----------------------------------|
| RETENTION, TAPE | | 3-16 |
| RETRIEVAL | | 11-8 , 11-9 , 11-12, 11-13, 11-17 |
| RETRIFVAL | | 11-20 |
| "RETURN | | 2-7 , 9-4 |
| RETURNS, MULTIPLE | | 4-6 |
| "REVERT | | 7-18 |
| REVISION | | II |
| REWIND | | 2-15 |
| *REWIND | | 7-3 |
| "REWIND | | 2-7 |
| REWIND, EDITLIB | | 7-10 |
| "RFL | | 1-14, 2-18 |
| RING, TAPE | | 3-16 |
| RIGS | | 11-12 |
| RIGSONLINE | | 11-12 |
| ROOT SEGMENT | | 2-19 |
| ROOTS, POLYNOMIAL | | 10-5 |
| ROUNT, FTN | | 4-2 |
| ROUTE | | 2-5 , 2-8 , 4-16, 9-9 , 10-5 |
| "ROUTE | | 2-10, 9-8 , 12-1 |
| RP, DEVICE SET | | 2-3 |
| RT, RECORD TYPE | | 3-18 |
| "RUN | | 4-1 |
| "RUN, COMPILER | | 9-4 |
| RUN, EDITOR | | 9-6 |
| RUN2P3 | | 10-1 |
| RVD, TAPE | | 2-21 |
| S | | PAGE |
| S-TAPE | | 3-17 |
| "SATISFY | | 2-15 |
| SAVE, EDITOR | | 9-6 , 9-14 |
| SAVE, SEGLOAD | | 2-19 |
| SCALE | | 10-7 |
| SCCALC | | 10-11 |
| SCIENTIFIC SUBROUTINE PACKAGE (IBM) | | 10-6 |
| SCOPE 3.3 | | 1-3 , 3-19, 4-11, 4-12 |
| SCOPE 3.4 | | 1-3 |
| SCORE | | 11-17 |
| SCORS | | 10-13 |
| SCRATCH FILES | | 2-7 , 3-1 , 3-13 |
| SCREEN | | 8-8 , 8-10 |
| SCREEN, INTERCOM | | 9-15 |
| SC4020 | | 10-13, 10-14, 10-24 |
| SC4020, CONVERT | | 10-11 |
| SC4060 | | 10-16, 10-17, 12-8 |
| SEARCH, PAGE | | 9-20 |

JUNE 1977

PAGE 15-21

| | S | (CONT) | PAGE |
|----------------------|---|--------|------------------------------|
| SECURITY | | | 3-10, 9-2 |
| SEEK | | | 3-2 |
| SEGLCAD | | | 4-25 |
| "SEGLCAD | | | 2-19, 4-10 |
| SEGMENTATION | | | 2-14, 2-19, 2-20, 4-10, 4-22 |
| SEGMENTATION | | | 5-12 |
| SELECT, COBOL | | | 5-4 |
| SEMICOLON | | | 1-10 |
| SEND | | | 13-1 |
| SEND | | | 9-13 |
| SEND, INTERCOM | | | 9-13 |
| SECTORAN | | | 6-1 |
| SECTORAN, EDITLIB | | | 7-11 |
| *SEQUENCE | | | 7-3 |
| SEQUENTIAL | | | 3-1, 3-2, 3-20, 7-2, 7-11 |
| SEQUENTIAL | | | 7-20 |
| SEQUENTIAL FILES | | | 6-1 |
| SETAL, EDITLIB | | | 7-10, 7-16 |
| *SETNAME | | | 3-11 |
| SEVEN-TRACK | | | 3-13 |
| SHARE | | | 10-23 |
| SHARP | | | 11-6, 11-13 |
| SHIFT | | | 4-5 |
| SHREDDER | | | 13-5 |
| SI TAPE | | | 3-13, 3-16, 3-17, 10-8 |
| SIGN OFF | | | 9-9 |
| SIGN-ON | | | 9-1 |
| SIGNON | | | 8-6 |
| SIGNCN, IGS | | | 8-2 |
| SIMI5 | | | 1-14 |
| "SIMI5, SIMI15 | | | 11-19 |
| SIMLIB | | | 10-1 |
| SIMPLE LOAD | | | 2-13 |
| SIMSCRIPT | | | 11-19 |
| SIMULATION | | | 11-12, 11-17 |
| SIS | | | 3-19, 3-20, 6-1 |
| "SISTAT | | | 3-19 |
| SITUATE, INTERCOM | | | 9-13 |
| *SKIP | | | 7-3 |
| SKIPB/SKIPF, EDITLIB | | | 7-10 |
| "SKIPF, SKIPB | | | 6-1 |
| SKPFIL | | | 4-10, 4-16 |
| "SLCAD | | | 2-15 |
| SLOT TAPE | | | 3-14, 3-15 |
| "SNOCOL | | | 10-21, 11-19 |
| SORT | | | 5-14, 7-22, 11-14, 13-5 |
| SORT/MERGE | | | 1-7 |

JUNE 1977

PAGE 15-22

| S | (CONT) | PAGE |
|---------------------------|--------|------------------------------|
| SORT/MERGE, FTN | | 4-12 |
| SORT, COBOL | | 5-4 |
| SORTER | | 13-5 |
| SORTING | | 10-5 |
| **SORTMRG | | 11-14 |
| SOURCE CODE | | 10-24 |
| SPECIAL FORMS | | 2-9, 14-3 |
| SPECIAL FUNCTIONS | | 10-5 |
| SPECIFICATION STATEMENTS | | 4-4 |
| SPLICE | | 3-14 |
| SPSS | | 11-12, 11-18 |
| SPY | | 2-7, 4-16 |
| SPYCNF, SPYOFF | | 4-16 |
| SRTLIB | | 10-1 |
| SSP | | 10-6, 10-7, 10-24 |
| STACK | | 3-2 |
| STAND ALONE 1700 | | 12-1 |
| STATIC | | 3-4 |
| STATISTICAL ANALYSIS | | 10-4 |
| STATISTICAL PROGRAMS | | 10-19 |
| STATISTICS | | 11-18 |
| STATUS | | 9-1 |
| STICKER LABEL | | 3-15 |
| STORAGE, TAPE | | 3-14 |
| STORE | | 9-14 |
| STRANGER TAPE | | 3-1, 3-13, 3-16, 10-14, 12-8 |
| STRESS | | 10-22 |
| STRING MANIPULATION | | 11-19 |
| STRING, PAGE | | 9-20 |
| STRUCTURAL ANALYSIS | | 11-11 |
| STRUCTURE | | 4-28 |
| SUBPROGRAM, COBOL (SUB) | | 5-2, 5-3 |
| SUBPROGRAMS, LIBRARIES OF | | 10-2 |
| SUBSCRIPT | | 5-9 |
| SUBSCRIPT BOUND | | 4-26 |
| SUBSCRIPTS | | 4-7, 5-5 |
| SUBSCRIPTS, COBOL | | 5-1 |
| SUGGESTION BOX | | 7-22 |
| **SUMMARY | | 2-7, 9-4, 9-9 |
| SUP, INTERCOM | | 9-1 |
| SWAP | | 1-4 |
| SYMBOL | | 10-7 |
| SYNCHRONOUS | | 8-8 |
| SYNTAX CHECK | | 5-15 |
| **SYSEULL | | 2-7, 9-9 |
| SYSIO | | 4-11, 10-1 |
| SYSMISC | | 4-11, 10-2 |

JUNE 1977

PAGE 15-23

S (CONT)

PAGE

| | |
|---------------------------|------------|
| SYSCVL | 10-1 |
| SYSTEM BULLETIN | 2-7 , 9-9 |
| SYSTEM LIBRARIES | 10-1 |
| SYSTEM PERMANENT FILES | 3-6 |
| SYSTEM STANDARD (SI) TAPE | 3-13, 3-14 |
| SYSTEM 2000 | 11-19 |
| SYSTEMC | 4-14 |

T

PAGE

| | |
|----------------------------|-----------------------------|
| T OPTION, FTN | 4-5 , 4-14 |
| TAB CHARACTER | 9-6 |
| TABULAR FORMAT (T) | 4-8 |
| TAPE | 1-4 , 2-3 , 2-7 , 2-21, 3-1 |
| TAPE | 3-15 |
| TAPE CHARACTER | 1-9 |
| TAPE CHARACTERISTICS | 3-13 |
| TAPE LIBRARIAN | 1-4 , 3-10, 3-16 |
| TAPE, STRANGER | 3-13 |
| TAPE, OFF | 9-5 |
| TAPE, ON | 9-3 , 9-5 |
| TAPES, STRANGER | 12-8 |
| TASK FILE | 8-2 |
| TEACH, INTERCOM | 9-4 |
| TEKTRONIX | 8-8 , 8-9 , 10-17 |
| TEK30 | 8-9 |
| TEK480, TEK300 | 10-18 |
| TELEPHONE, COMPUTER ACCESS | 1-1 , 1-2 , 9-1 |
| TELETYPE | 1-3 , 9-1 |
| TERMINAL | 12-1 |
| TERMINAL IDENTIFICATION | 2-11 |
| TEXAS INSTRUMENTS | 12-15 |
| TEXT EDITOR | 9-21 |
| TEXT PROCESSING | 12-16 |
| TEXT REPLACEMENT, EDITOR | 9-7 |
| TEXT SCANNING | 11-20 |
| THREE-D | 10-12 |
| TID, ROUTE | 2-11 |
| TIME | 2-3 |
| TIME LIMIT | 9-4 , 9-15, 14-1 |
| TIME SHARING | 12-13 |
| TIME-SHARING FTN (TS) | 4-2 , 4-13 |
| TIP | 12-16, 12-17 |
| TRACE | 4-15 |
| TRACE, COBOL | 5-1 |
| TRACE, FTN | 4-2 |
| TRACING (T) | 4-26 |
| "TRANSF | 2-7 |

JUNE 1977

PAGE 15-24

| T (CONT) | PAGE |
|---------------------------|------------------------------|
| "TRANSPF | 3-12 |
| TREE | 2-19 |
| TRCUPLE FORM | 13-1 |
| TRUE | 1-8 |
| TSKLCAD | 8-4 |
| TURNKEY (TK) | 3-6 |
| TURNKEY, INTERCOM | 9-1 , 9-2 |
| TUTORIAL | 11-6 |
| TYPE, REAL | 4-5 |
| U | PAGE |
| U, COBOL | 5-2 |
| UNBLOCKED | 3-1 |
| UNBLOCKED FILES | 4-11 |
| UNCLASSIFIED | 3-3 |
| UNIT | 4-20 |
| UNIT RECORD | 1-3 , 3-2 , 4-8 , 9-18 |
| "UNLOAD | 2-7 |
| UNLOAD | 4-16 |
| UNNAMED PROCEDURES | 7-18, 7-20 |
| UNSATISFIED EXTERNALS | 4-15 |
| UNSATISFIED REFERENCE | 4-6 |
| "UPDATE | 7-2 |
| UPDATE DIRECTIVES | 7-3 |
| USE, USER LOADER | 2-16 |
| USER ID | 3-4 |
| USER INITIALS | 2-3 , 2-4 , 9-1 |
| USER LIBRARY | 7-9 |
| USER NAME | 9-1 |
| USER RETURN CODES | 3-7 |
| USER SERVICES | 13-1 |
| USER TROUBLE FORM (UTF) | 13-1 |
| USERS FORUM | 13-1 |
| USING, COBOL | 5-3 , 5-5 |
| UTILITIES | 3-19, 4-16, 6-1 , 7-1 , 13-5 |
| UTILITY | 10-20 |
| UTILITY | 10-19, 10-21, 10-24 |
| UTILITY | 7-22 |
| V | PAGE |
| VECTOR | 10-5 |
| VENUS | 11-20 |
| VIM | 10-22, 10-23 |
| VISUAL REEL NUMBER (VRNO) | 3-16 |
| VOLUME SERIAL NUMBER | 3-10, 3-15 |
| VSN | 3-17 |
| "VSN | 3-13, 3-15, 4-11 |

| | | | |
|---------------------------|-------|--------|-------------------------------|
| | V | (CONT) | PAGE |
| VSN, VOLSER | | | 3-10 |
| | W | | PAGE |
| W, TAPE LABEL | | | 3-16 |
| WALL CLOCK | | | 3-13 |
| WASHINGTON, UNIVERSITY OF | | | 6-4 , 7-18 |
| *WFCR | | | 7-3 |
| WORD LENGTH | | | 1-7 |
| WORKING-STORAGE, COBOL | | | 5-3 |
| WRITMS | | | 4-10, 4-12 |
| | X | | PAGE |
| X-OFF | | | 9-3 , 9-5 |
| XEQ, INTERCOM | | | 4-11, 9-7 , 9-16, 9-16 |
| | Y | | PAGE |
| *YANK | | | 7-3 |
| | Z | | PAGE |
| Z PARAMETER | | | 5-7 |
| Z-TYPE RECORDS | | | 9-18 |
| Z, COBOL | | | 5-2 |
| ZERO-BYTE TERMINATED | | | 5-7 |
| ZIP MODE | | | 12-11 |
| ZPFUNC | | | 4-16, 10-5 |
| ZZZZZ FILES | | | 9-3 |
| ZZZZZEF | | | 3-19 |
| | 0 - 9 | | PAGE |
| 026 | | | 1-6 , 1-9 , 12-3 , 13-5 |
| 029 | | | 1-6 , 1-9 , 2-12, 12-3 , 13-5 |
| 1700 | | | 1-6 , 8-1 , 10-8 |
| 1700 COMPUTER | | | 12-1 |
| 200 USER TERMINAL | | | 1-6 , 12-3 |
| 274 | | | 8-1 |
| 274CALCOMP | | | 8-5 |
| 4015, TEKTRONIX | | | 8-8 |
| 4020 | | | 10-13, 10-14 |
| 4060 | | | 10-13, 10-16, 12-8 |
| 4530 COM | | | 12-9 |
| 4662, TEKTRONIX | | | 8-9 |
| 60, SORTMRG | | | 11-14 |
| 6400 | | | 1-2 |
| 6600 | | | 1-2 |
| 6638 DISK | | | 3-2 |
| 6700 | | | 1-1 |

JUNE 1977

PAGE 15-26

0 - 9 (CONT)

PAGE

7-TRACK TAPE
763 CALCOMP
844 DISK
9-TRACK TAPE
936 CALCOMP

3-16
10-8 , 12-11
1-3 , 3-2
3-16, 6-9 , 12-3 , 12-15
10-8 , 12-10

SPEC

PAGE

\$ DELIMITER
\$, DUP
*/ COMMENT

2-20
7-19
7-11

INITIAL DISTRIBUTION

Copies:

12 Director
 Defense Documentation Center (TIMA)
 Cameron Station
 Alexandria, Virginia 22314

CENTER DISTRIBUTION

Copies:

| | | |
|-----|--------|------------------|
| 1 | 1800 | Gleissner G H |
| 1 | 1804 | Avrunin L |
| 1 | 1805 | Cuthill E H |
| 1 | 1809 | Wright C |
| 1 | 1809.3 | Harris D |
| 1 | 1820 | Camara W |
| 1 | 1840 | Lugt H J |
| 1 | 1850 | Corin T |
| 1 | 1860 | Sulit R A |
| 1 | 1890 | Gray G R |
| 1 | 189.1 | Taylor N M |
| 1 | 189.2 | Hayden H P |
| 1 | 1891 | Cooper A E |
| 960 | 1892.1 | Strickland J D |
| 10 | 1892.2 | Sommer D V |
| 1 | 1892.3 | Minor L R |
| 1 | 1894 | Seals W |
| 1 | 1896 | Blackburn P |
| 1 | 1896.2 | Dennis L |
| 1 | 522 | Library Division |